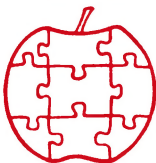


Apple

\$1.50



Assembly

Line

Volume 2 -- Issue 11

August, 1982

In This Issue...

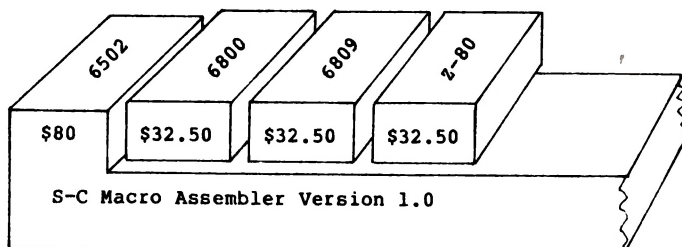
Search and Perform Subroutine	2
Auto-Manual Toggle Patch for S-C Macro Assembler	6
Improved DOS Free Space Patch	9
Videx 80-Column Patches for S-C Macro Assembler	11
Review of "Apple Graphics & Arcade Game Design"	23
Quick Way to Write DOS on a Disk	24
Correction to Relocatable JSR Article (July 1982)	24
Efficient Handling of Very Large Assembly Source Files	25
Lower Case in .AS and Literal Constants	28
Blinking Underscore Cursor	29
Review of QUICKTRACE	32

Subscription Renewals

If your address label shows a number 8209 or smaller in the upper right corner, it is time to renew. That is \$15 bulk mail in the USA; \$18 First Class in USA, Canada, and Mexico; \$28 to other countries.

New Macro Cross Assemblers Available

There are now three cross-assembler modules ready for the S-C Macro Assembler, and more to come. Each cross-assembler disk costs \$32.50 to registered owners of the S-C Macro Assembler. You get both regular and language card versions, with documentation of the special features and differences.



Search and Perform Subroutine.....Bob Sander-Cederlof

When writing an editor or other single-keystroke command system, a very common need is a subroutine which branches according to the value of a character. In Pascal and some other languages there is even a special statement for this programming need: CASE. You might do it like this in Applesoft:

```
1000 GET A$
1010 IF A$ = "A" THEN 2000
1020 IF A$ = "C" THEN 3000
1030 et cetera
```

You will often find the equivalent code in assembly language programs:

```
1000      LDA CHARACTER
1010      CMP #'A
1020      BEQ CHAR.WAS.A
1030      CMP #'C
1040      BEQ CHAR.WAS.C
1050      et cetera
```

Of course, it frequently happens that the number of different values is small, and the code sequence above with several CMP-BEQ pairs is the most efficient. It loses a little of its appeal, though, when you have to do it for more than about ten different values. And what if the branch points are too far away for BEQ relative branches? Then you have to write:

```
1000      LDA CHARACTER
1010      CMP #'A
1020      BNE .1
1030      JMP CHAR.WAS.A
1040 .1     CMP #'C
1050      BNE .2
1060      JMP CHAR.WAS.C
1070 .2     et cetera
```

That takes seven bytes of program for each value of the character.

Personally, I like to put the possible values and the corresponding branch addresses in a table, and search that table whenever necessary. Each table entry takes only three bytes. If the subroutine is used with several tables, and if there are a lot of possible values, then the tabular method saves a lot of memory.

I used the tabular method in my still-in-development word-processor. To speed and simplify the coding of the table entries, I wrote a macro definition JTBL as follows:

```
1020      .MA JTBL
1030      .DA #$1,12-1
1040      .EM
```


This defines a macro JTBL with two parameters. The first one will be the hexadecimal value to compare the test-character with, and the second one will be the branch address for that value. For example, if I write the macro call:

```
1400      >JTBL 86,FLIP.CHARS
```

the S-C Macro Assembler will generate:

```
.DA #$86,FLIP.CHARS-1
```

The "-1" is appended to each branch address in the table, because I use the PHA-PHA-RTS method to perform the branch. Before I go any farther, here is the search and branch subroutine:

```
1220 SEARCH.AND.PERFORM.NEXT
1230     INY                POINT TO NEXT ENTRY
1240     INY
1250     INY
1260 SEARCH.AND.PERFORM
1270     LDA T.BASE,Y       GET VALUE FROM TABLE
1280     BEQ .1            NOT IN THE TABLE
1290     CMP CURRENT.CHAR
1300     BNE SEARCH.AND.PERFORM.NEXT
1310 .1     LDA T.BASE+2,Y  LOW-BYTE OF BRANCH
1320     PHA
1330     LDA T.BASE+1,Y     HIGH-BYTE OF BRANCH
1340     PHA
1350     LDY #0             (SINCE MOST BRANCHES WANT Y=0)
1360     RTS                DO THE BRANCH!
```

There are so far four different value-branch tables in my word processor. Here is an abbreviated listing:

```
1380 T.BASE
1390 T.ESC0 >JTBL 81,AUXILIARY.MENU
1400      >JTBL 82,SCAN.BEGIN
1410      >JTBL 83,TOGGLE.CASE.LOCK
. . . . .
1540      >JTBL 9B,ESC0.ESC
1550      >JTBL 00,SC.BELL
1560 *-----
1570 T.ESC2 >JTBL 81,AUXILIARY.MENU
. . . . .
1690      >JTBL EB,SCAN.RIGHT
1700      >JTBL ED,SCAN.DOWN
1710      >JTBL 00,ESC2.END
1720 *-----
1730 T.MAIN >JTBL C4,MAIN.DOS
1740      >JTBL C5,MAIN.EDIT
. . . . .
1800      >JTBL D3,MAIN.SAVE
1810      >JTBL 00,MON.BELL
1820 *-----
1830 T.AUX  >JTBL C3,COPY.BLOCK
1840      >JTBL C4,DELETE.BLOCK
. . . . .
```

```

1890          >JTBL D3,SAVE.SEGMENT
1900          >JTBL 00,SC.BELL

```

Notice that each of the four tables ends with a 00 value. The branch address after the 00 value tells where to branch if the current character does not match any values in the table.

When I want to compare the current character with entries in the T.MAIN table, here is how I do it:

```

2000          LDY #T.MAIN-T.BASE
2010          JSR SEARCH.AND.PERFORM

```

The LDY instruction sets Y to the offset of the table from T.BASE, and the search subroutine references the table relative to T.BASE. I use JSR to call the search subroutine. The search subroutine uses PHA-PHA-RTS to effectively JMP to the chosen branch address. And then the value processor ends with RTS to return to the next line after the JSR SEARCH.AND.PERFORM.

Counting all four tables, I have 45 branches, occupying $3 \times 45 = 135$ bytes. If I had used the CMP-BEQ method, which occupy four bytes per value, it would have taken $4 \times 45 = 180$ bytes. The subroutine is only 23 bytes long, so I saved 22 bytes. But if I needed the longer CMP-BNE-JMP sequences throughout, I would have had $7 \times 45 = 315$ bytes! Wow! Long live tables!

Tables have even more advantages. For one, they are a lot easier to modify when you want to add or delete a value. For another, the program is easier to read when there is no rat's nest of branches to try to unravel. For me, it almost makes the assembly listing as easy to read as the reference manual!

Notice that it would be possible to overlap tables using my subroutine. I might need at some times to search for 13 different values, and at others to search for only 7 of those same values, with the same branches. If so, the seven entries in common would be grouped at the end of the 13-entry table. The table has two labels, like this:

```

3000 T.13      >JTBL C1,DO.A
3010           >JTBL C4,DO.D
      . . . .
3050           >JTBL CF,DO.O
3060 T.7       >JTBL C2,DO.B
3070           >JTBL C5,DO.E
      . . . .
3120           >JTBL D7,DO.W
3130           >JTBL 00,DO.NOTHING

```

What about speed? Well, it is pretty fast too. The CMP-BNE-JMP takes five cycles for each value that does not compare equal, and finally seven cycles for the one which compares equal. If the tenth comparison bingos, that is $9 \times 5 + 7 = 52$ cycles. The subroutine takes 171 cycles for the same search. Over three times longer, but still less than 120 microseconds longer. You would have to perform the search over 8000 times in one day to add a whole second of computer time!

Here is a small program to accompany Bill Morgan's Automatic Catalog in the June '82 issue of AAL. This routine adds an AUTO/MANUAL command toggle to the S-C Macro Assembler. Using CTRL-A when the cursor is at the beginning of a line enters the AUTO line numbering mode and waits for input of a line number and/or RETURN. Entering another CTRL-A while in AUTO mode and at the start of a line executes a MANUAL command.

In addition, I have added some code to provide slow and fast listings at a single keypress. CTRL-S does a SLOW LIST command, which is cancelled by a 'RETURN' during listing. CTRL-L will provide a listing at normal speed (assuming the slow list has been cancelled.)

The patch is implemented as follows:

1. Enter the S-C Macro Assembler
2. :\$101D:33 N 1000G
3. :BLOAD AUTO/MANUAL PATCH
4. :\$138D: 4C 28 32 (JMP PATCH instead of JSR BELL)
5. :BSAVE AUTO/MAN S-C MACRO ASM,A\$1000,L\$2300

Note: You may omit step 2 if you have already installed Bill's automatic CATALOG.

OFTEN WONDER HOW MACHINE LANGUAGE PROGRAMS WORK?

Well stop wondering and do something about it! Use DISASM to convert 6502 machine code into meaningful, symbolic source. Create a text file which is directly compatible with DOS ToolKit, LISA and S-C (both 4.0 & Macro) Assemblers. DISASM handles data tables, displaced object code and even lets you substitute MEANINGFUL labels of your own choice (100 commonly used Monitor & Pg Zero names included in Source form to get you rolling). An address-based cross reference table provides even more insight into the inner workings of machine language programs. DISASM is an invaluable aid for both the novice and expert alike.

DISASM (Version 2.2): \$30.00

The 'MIRROR': Firmware for Apple-Cat

Communications ROM plugs directly into Novation's modem card. Three basic modes: Dumb Terminal, Remote Console & Programmable Modem. Added features include: Printer buffer, Pulse or Tone dialing, true dialtone detection, audible ring detect and ring-back option. Supports VIDEK 80-column board and Apple's Comm card commands. (Hardware differences prevent 100% interchangeability with Comm card.)

ROM & User's Manual: \$29.00

Utilities For Your S-C Assembler (4.0)

SC.GSR: A Global Search and Replace Eliminates Tedious Manual Renaming Of Labels..... \$20.00

SC.XREF: A Linenumber-Based Global Cross Reference Table For Complete Source Documentation... \$20.00

SC.TAB: Tabulates Source Files Into Neat, Readable Form. Encourages Fast, Free-Format Entry. \$15.00

SC UTILITY PAK: Includes All Three Utilities Described Above (You Save \$10.00)..... \$45.00

Avoid A \$3.00 Shipping/Handling Charge By Mailing Full Payment With Order

R A K - W A R E
41 Ralph Road
West Orange NJ 07052

***** SAY YOU SAW IT IN 'APPLE ASSEMBLY LINE'! *****

```

1000 *-----
1010 * AUTO/MANUAL TOGGLE
1020 *
1030 * BY ROBERT F. O'BRIEN
1040 * 14, CLONSHAUGH LAWN, DUBLIN 5.
1050 *-----
1060 .OR $3228
1070 .TF AUTO/MANUAL PATCH
1080 *-----
0024- 1090 CH .EQ $24
11D2- 1100 SC.SLOW .EQ $11D2
135E- 1110 SC.REENTER .EQ $135E
13C3- 1120 SC.RETURN .EQ $13C3
152A- 1130 SC.INSTALL .EQ $152A
183F- 1140 SC.LIST .EQ $183F
FF3A- 1150 MON.BELL .EQ $FF3A
1160 *-----
3228- C9 81 1170 AUTO.MANUAL.COMMAND
322A- F0 0E 1180 CMP #$81 CTRL-A?
322C- C9 8C 1190 BEQ AUTO.TOGGLE
322E- F0 41 1200 CMP #$8C CTRL-L?
3230- C9 93 1210 BEQ LIST
3232- F0 49 1220 CMP #$93 CTRL-S?
1230 BEQ SLOW.LIST
1240 *
3234- 20 3A FF 1250 BACK JSR MON.BELL
3237- 4C 5E 13 1260 JMP SC.REENTER BACK TO ASSEMBLER
1270 *-----
323A- A5 24 1280 AUTO.TOGGLE
323C- C9 01 1290 LDA CH
323E- F0 06 1300 CMP #1 BEGINNING OF LINE?
3240- C9 06 1310 BEQ AUTO.CMD
3242- F0 16 1320 CMP #6 AFTER LINE NUMBER?
3244- D0 EE 1330 BEQ MANUAL.CMD
1340 BNE BACK
1350 *-----
3246- A2 00 1360 AUTO.CMD
3248- BD 55 32 1370 LDX #0
324B- 20 2A 15 1380 .1 LDA AUTO.TEXT,X GET CHARACTER
324E- E0 05 1390 JSR SC.INSTALL PROCESS CHAR
3250- 90 F6 1400 CPX #5
3252- 4C 5E 13 1410 BCC .1
3255- C1 D5 D4 1420 JMP SC.REENTER
3258- CF A0 1430 AUTO.TEXT .AS -/AUTO /
1440 *-----
325A- A2 00 1450 MANUAL.CMD
325C- 86 24 1460 LDX #0
325E- BD 6B 32 1470 STX CH GO TO START OF LINE
3261- 20 2A 15 1480 .1 LDA MANUAL.TEXT,X
3264- E0 06 1490 JSR SC.INSTALL
3266- 90 F6 1500 CPX #6
3268- 4C C3 13 1510 BCC .1
326B- CD C1 CE 1520 JMP SC.RETURN
326E- D5 C1 CC 1530 MANUAL.TEXT .AS -/MANUAL/
1540 *-----
3271- A5 24 1550 LIST LDA CH
3273- C9 01 1560 CMP #1 BEGINNING OF LINE?
3275- D0 BD 1570 BNE BACK NO, RETURN TO ASSEMBLER
3277- 20 3F 18 1580 JSR SC.LIST
327A- 4C C3 13 1590 JMP SC.RETURN
1600 *-----
327D- A5 24 1610 SLOW.LIST
327F- C9 01 1620 LDA CH
3281- D0 B1 1630 CMP #1
3283- 20 D2 11 1640 BNE BACK
3286- 20 3F 18 1650 JSR SC.SLOW SET SLOW MODE
3289- 4C C3 13 1660 JSR SC.LIST
1670 JMP SC.RETURN

```

QUICKTRACE

relocatable program traces and displays the actual machine operations, while it is running without interfering with those operations. Look at these **FEATURES**:

Single-Step mode displays the last instruction, next instruction, registers, flags, stack contents, and six user-definable memory locations.

Trace mode gives a running display of the Single-Step information and can be made to stop upon encountering any of nine user-definable conditions.

Background mode permits tracing with no display until it is desired. Debugged routines run at near normal speed until one of the stopping conditions is met, which causes the program to return to Single-Step.

QUICKTRACE allows changes to the stack, registers, stopping conditions, addresses to be displayed, and output destinations for all this information. All this can be done in Single-Step mode while running.

Two optional display formats can show a sequence of operations at once. Usually, the information is given in four lines at the bottom of the screen.

QUICKTRACE is completely transparent to the program being traced. It will not interfere with the stack, program, or I/O.

QUICKTRACE is relocatable to any free part of memory. Its output can be sent to any slot or to the screen.

QUICKTRACE is completely compatible with programs using Applesoft and Integer BASICs, graphics, and DOS. (Time dependent DOS operations can be bypassed.) It will display the graphics on the screen while **QUICKTRACE** is alive.

QUICKTRACE is a beautiful way to show the incredibly complex sequence of operations that a computer goes through in executing a program

QUICKTRACE

\$50

Is a trademark of Anthro-Digital, Inc.

Copyright © 1981

Written by John Rogers

See these programs at participating Computerland and other
fine computer stores.

Anthro - Digital Software, Inc.
P.O. Box 1385 Pittsfield, MA 01202

Free Space Patch For the S-C Assembler.....Mike Sanders

Volume 5, Number 6 of Call A.P.P.L.E. has an article giving a DOS patch to replace the volume number printed during catalog with number of free sectors remaining on the disk.

The routine as published works for both Applesoft and Integer BASIC, but does not work with the language card version of the S-C Assembler. Only a few changes were needed to make it work with all three.

A call to Bob gave me the location of the decimal print routine in the S-C Macro Assembler, Language Card Version.

The original code as published in Call A.P.P.L.E. checked location \$E006 to see what language is in use. My code looks at \$E001, which has a different value in each of the three:

Language	\$E001
Applesoft:	\$28
Integer BASIC:	\$00
S-C Macro Assembler:	\$94

The code in lines 1320-1370 checks which language is in use and jumps to the right routine. I also changed the zero page locations used to count the number of free sectors because the S-C Assembler print routine expects the two-byte value to be in \$D3 and \$D4.

The rest of the code works as explained in the Call A.P.P.L.E. article. I refer you to it for more details and as an excellent lesson on reducing the size of code.

Install the two patches to DOS by BLOADing the two binary files FREE.SECTORS.1 and FREE.SECTORS.2. The type CATALOG to see the how many free sectors you have.

```
1000 *SAVE S.FREE SECTORS
1010 *-----
1020 *          FREE SECTORS PATCH FOR DOS 3.3
1030 *-----
00D3- 1040 LOBYTE .EQ $D3
00D4- 1050 HIBYTE .EQ $D4
1060 *-----
B3F2- 1070 SECTOR.MAP .EQ $B3F2
E001- 1080 LANG.ID .EQ $E001      LANGUAGE ID
E51B- 1090 PRT.INT .EQ $E51B   INTEGER BASIC PRINT ROUTINE
ED24- 1100 PRT.FP .EQ $ED24   APPLESOFT PRINT ROUTINE
DE00- 1110 PRT.SC .EQ $DE00    S-C ASSEMBLER PRINT ROUTINE
1120 *-----
1130          .OR $BA69
1140          .TF FREE.SECTORS.1
1150 *-----
1160 FREE.SECTOR.PATCH
BA69- A0 C8      1170 LDY #$C8
BA6B- B9 F2 B3  1180 .1 LDA SECTOR.MAP,Y
BA6E- F0 0D      1190 BEQ .4          NO FREE SECTORS IN THIS BYTE
BA70- 0A         1200 .2 ASL          SHIFT INTO CARRY
BA71- 90 FD      1210 BCC .2          SECTOR IN USE
```

BA73- 48	1220	PHA	SECTOR FREE
BA74- E6 D3	1230	INC LOBYTE	COUNT IT
BA76- D0 02	1240	BNE .3	
BA78- E6 D4	1250	INC HIBYTE	
BA7A- 68	1260 .3	PLA	SECTOR MAP BYTE AGAIN
BA7B- D0 F3	1270	BNE .2	IF ANY LEFT
BA7D- 88	1280 .4	DEY	NEXT BYTE OF SECTOR MAP
BA7E- D0 EB	1290	BNE .1	
BA80- A6 D3	1300	LDX LOBYTE	VALUE IN X AND A
BA82- A5 D4	1310	LDA HIBYTE	
BA84- AC 01 E0	1320	LDY LANG.ID	CHECK WHICH LANGUAGE
BA87- 30 08	1330	BMI SCASM	\$94: S-C ASSEMBLER
BA89- F0 03	1340	BEQ INTEGR	\$00: INTEGER BASIC
BA8B- 4C 24 ED	1350	JMP PRT.FP	\$28: APPLESOFT
BA8E- 4C 1B E5	1360	INTEGR JMP PRT.INT	
BA91- 4C 00 DE	1370	SCASM JMP PRT.SC	
	1380	*-----	
	1390	.OR \$ADB9	
	1400	.TF FREE.SECTORS.2	
	1410	*-----	
ADB9- EA	1420	NOP	FILLER
ADBA- A9 00	1430	LDA #0	ZERO THE COUNT
ADBC- 85 D3	1440	STA LOBYTE	
ADBE- 85 D4	1450	STA HIBYTE	
ADCO- 20 69 BA	1460	JSR FREE.SECTOR.PATCH	

DO YOU OWN ONE OF THOSE SMART PRINTERS?

(But Are Using It With A 'Dumb' Interface Board?)

Now you can get the most out of your EPSON, NEC, C.Itoh and OKI printers with the PERFORMER board for the Apple II and Apple II Plus. This board plugs into any Apple slot and turns your 'dumb' printer interface into a 'smart' one. Here's an example set-up menu for the NEC 8023A:

PICA	ON	\$ Easy to use! Menu-driven with simple commands
ELITE	OFF	\$ Replaces tedious manual printer set-up
CONDENSED	OFF	\$ No need to remember those 'ESC' command sequences
ENLARGED	OFF	\$ The PERFORMER is in ROM so its always 'on-line'
ENHANCED	OFF	\$ Easy selection of available printer fonts
LINES/INCH	SIX	\$ Also controls print format with dynamic defaults
PAGE NO.	1	\$ Defaults are easily overridden for maximum versatility
COLUMNS	80	\$ Optional Header line prints Title, Date & Pg
INDENT	0	\$ Provides Pgl/Pg 2 TEXT or GRAPHICS screen dumps
FORM LENGTH	66	\$ Large format graphics in Positive or Negative images
LINES/PAGE	63	\$ Compatible with Apple, Tmac, Epson, Microtek and
FORM FEED	ON	\$ similar 'dumb' Centronics type parallel I/F boards
DISPLAY	OFF	\$ SPECIFY printer: EPSON MX80 W/Graftrax-80
GRAPHICS	POS	\$ EPSON MX100, EPSON MX80/MX100 W/Graftrax Plus
DUMP	P61	\$ NEC 8023A, C.Itoh 8510 (ProWriter)
		\$ OKI Microline 82A/83A W/OKIGRAPH

PERFORMER BOARD: \$49.00

Avoid A \$3.00 Shipping/Handling Charge By Mailing Full Payment With Order

R A K - W A R E
41 Ralph Road
West Orange NJ 07052

The Macro-Videx Connection.....Don Taylor

It seems that whenever I purchase a new hardware product for my Apple, I spend countless hours honing my most precious software tools to make them compatible with it. I purchased my Videx Videoterm card for use with Pascal, and had no intention of using it with the S-C Assembler. Then one fateful day I made a temporary patch to Version 4.0 -- just to see what it would look like -- and I was immediately hooked....

You won't believe what it's like to assemble with 80 columns of display! You can actually write source files that are legible on the screen, with no wraparound on comments -- even during assembly. What you see on the display is what you would see on a printer, only cleaner.

When I upgraded to the S-C Macro Assembler, I was compelled to produce a configuration file that would modify the new assembler to work with the Videoterm board. The resulting source file is included with this article.

The assembled SCM80 file will reconfigure a copy of the S-C Macro Assembler Version 1.0 that is currently resident in memory (for more about this concept, see "Controlling Software Configuration", AAL April '82).

Once the mods are installed you will be able to use your Videx for everything except: (1) Using the Escape-L sequence to LOAD a disk file whose name appears on the display, and (2) Using the copy key (right arrow). You will still be able to use Escape-L to generate the normal dashed comment line, and you can use the other escape functions to move the cursor and clear portions of the screen.

SCM80 will display control characters (and other selected strings intended to be so) in inverse on your screen, provided you have the standard (inverse) alternate character generator ROM installed in your Videoterm. If you have some other ROM installed, these characters and strings may be printed in Chinese. In this case you may want to modify the new character output routine!

SCM80 will also permit painless switching of case while using the assembler. A control-A keypress will always be recognized as a "shift lock" signal, while a control-Z will be treated as a "shift unlock". This feature makes it easy to write easy-to-read source files.

The assembled SCM80 code is moved into memory immediately following the assembler, and is located at one of two places, depending on which flavor (vanilla or language card) of the assembler you're using. The flavor of the configuration file is made to match that of the assembler through the use of a conditional flag (LCVERSION) and several conditional assembly statements. Another equate variable, SLOTNUM, allows you to specify the slot in which your Videx board resides.

APPLE PERIPHERALS ARE OUR ONLY BUSINESS

TIME II

THE MOST POWERFUL, EASIEST TO USE CLOCK FOR YOUR APPLE

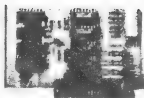
- Time in hours, minutes and seconds.
- Date with year, month, day of week and leap year.
- Will enhance programs for accounting, time and energy management, remote control of appliances, laboratory analysis, process control, and more.
- 24-hour military format or 12-hour with AM/PM indication.
- User selectable interrupts permit foreground/background operation of two programs simultaneously.
- Crystal controlled for .0005% accuracy.
- Easy programming in basic.
- On board battery backup power for over four months power off operation (battery charges when Apple is on).



- Twenty-seven page operating manual included with many examples of programs to use with your Apple in any configuration.
- Includes disk containing a DOS Dater and many other time oriented utilities plus over 25 user contributed programs at no extra cost.

PRICE \$129.00

SUPER MUSIC SYNTHESIZER



- Complete 16 voice music synthesizer on one card. Just plug it into your Apple, connect the audio cable (supplied) to your stereo and boot the disk supplied and you are ready to input and play songs.
- It's easy to program music with our compose software. You will start right away at inputting your favorite songs. The Hi-Res screen shows what you have entered in standard sheet music format.

- We give you lots of software. In addition to Compose and Play programs, the disk is filled with songs ready to run.
- Easy to program in basic to generate complex sound effects.
- Four white noise generators which are great for sound effects.
- Plays music in true stereo as well as true discrete quadraphonic.
- Envelope control.
- Will play songs written for ALF synthesizer (ALF software will not take advantage of all the features of this board. Their software sounds the same in our synthesizer.)
- Automatic shutoff on power-up or if reset is pushed.
- Many many more features.

PRICE \$159.00

ANALOG TO DIGITAL CONVERTER

- 8 Channels
- 8 Bit Resolution
- On Board Memory
- Ratiometric Capability
- Fast Conversion (.078 ms per channel)
- Eliminates The Need To Wait For A/D Conversion (just PEEK at data)
- A/D Process Totally Transparent to Apple (looks like memory)

The analog to digital conversion takes place on a continuous, channel sequencing basis. Data is automatically transferred to on board memory at the end of each conversion. No A/D converter could be easier to use.

Our A/D board comes standard with 0, 10V full scale inputs. These inputs can be changed by the user to 0, -10V, or -5V, +5V or other ranges as needed.

The user connector has + 12 and -12 volts on it so you can power your sensors. (These power sources can be turned off with on board dip switch).

Accuracy 0.3% Input Resistance 20K Ohms Typ
A few applications may include the monitoring of • flow • temperature • humidity • wind speed • wind direction • light intensity • pressure • RPM • soil moisture and many more.

PRICE \$129.00

DIGITAL INPUT/OUTPUT BOARD

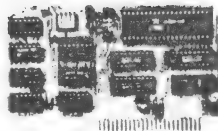
- Provides 8 buffered outputs to a standard 16 pin socket for standard dip ribbon cable connection.
- Power-up reset assures that all outputs are off when your Apple is first turned on.
- Features 8 inputs that can be driven from TTL logic or any 5 volt source.
- Your inputs can be anything from high speed logic to simple switches.

- Very simple to program, just PEEK at the data.
- 4 other outputs are also provided. User 1, reset, interrupt request, non-maskable interrupt.
- Now on one card, you can have 8 digital outputs and 8 digital inputs each with its own connector. The super input/output board is your best choice for any control application.

PRICE \$82.00

Z-80 CARD

- TOTALLY compatible with all CP/M software.
- Executes the full Z-80 and 8080 instruction set.
- Allows you to run your Apple CP/M based programs.
- Does EVERYTHING the other Z-80 boards do, plus supports Z80 Interrupts.
- Hardware and software settable switch options.
- An on-card PROM eliminates many I.C.'s for a cooler, less power consuming board.
- Complete documentation included. (user must furnish software)



PRICE \$139.00

Since our inception, Applied Engineering has continually expanded its line of Apple peripherals bringing you easy-to-use designs. We are the innovators not the imitators. Utilizing state-of-the-art technologies, Applied Engineering is continually improving its products. The above represents our most recent development. Applied Engineering offers you the highest quality peripherals at the lowest possible price. Applied Engineering's products are fully tested with complete documentation and available for immediate delivery. All products are guaranteed with a one year warranty.

All Orders Shipped Same Day.
Texas Residents Add 5% Sales Tax.
Add \$10.00 If Outside U.S.A.

Send Check or Money Order to:
APPLIED ENGINEERING
P.O. Box 470301
Dallas, TX 75247

See Your Dealer
or Call (214) 482-2027
7 Days a Week
Master Card & Visa Welcome

How It All Works

There are two primary steps involved in installing the modified code in the assembler: (1) Moving the new code into the area of memory immediately following the assembler, and (2) Patching the existing assembler code to point to the new routines and then returning or cold-starting the system.

The SCM80 code contains both the new Videoterm support routines and the routines used to install those support routines. It loads in at \$4000, stuffs the Videoterm routines just beyond the assembler code, and then performs the return or cold start. Depending on the flavor, a few other small tasks are performed in the process; let's take a closer look.

Lines 1280-1310 contain the two constants used to tailor SCM80 to assembler flavor and Videoterm slot number. The last two lines are the starting addresses where the new code will be relocated, depending on the flavor. The LCVERSION flag is used to determine the base address of the assembler in lines 1340-1380; this base address is used throughout the rest of the listing to determine absolute patch addresses within the assembler.

The Videoterm support routines are contained in lines 3240-3770. Lines 3400-3700 contain replacement routines for two of the routines in the line editor portion of the assembler. The NEW.WARM.ENTRY routine in lines 3240-3260 is intended to keep the Videoterm in the saddle during a RESET or system warm start.

The code in lines 3820-4740 are replacements for some of the standard monitor routines. Several of these routines have no other purpose than to support the escape cursor movements. In the case of the language card flavored RDKEY, an extra subroutine is provided to unprotect the RAM during case-shift sequences (more about that in a minute).

Lines 1770-2040 use the monitor's MOVE routine to slip the support routines into their designated origin at \$3200 or \$F400. The vanilla version patches the assembler's symbol table address to make room for the move; the language card version unprotects RAM prior to the move.

The patching of the assembler is done in lines 2050-2920. unused code is NOP-ed out here, and jumps are strategically poked in to point to the new routines. A replacement escape jump table created in lines 2950-3090 gets installed in the assembler, so the new escape routines can be accessed in the standard manner. The assembler's cold start routines are patched to point to the resilient NEW.WARM.ENTRY routine (more about that in a moment, too).

Lines 2870-2920 complete the installation and patching process. For the vanilla version, a simple RTS returns control to the calling program. The language card version first write protects RAM and then performs a DOS cold start. Once the assembled code has been installed and the patches made, the

installation portion of SCM80 is of no use, so a cold start should be performed to reset the assembler's file pointers, leaving only the SCM80 code that is now supporting your Videoterm.

Assembly and Installation

You'll note the absence of any .TF directive in the listing, meaning you'll have to manually save this file when you're done. This is because although the resulting object code will be located in continuous memory, it has origins (.OR directives) at two locations. The actual length of the file is calculated by a variable called LENGTH. The instructions for assembly are contained in the source file's title block. I call my vanilla patch file SCM80. and the language card version SCM80.LC.

With the assembler code resident in memory, there are several ways of installing the patches. Perhaps the most straightforward is to BRUN the assembled patch file, or BLOAD it and type 4000G as a monitor command. If you're using the vanilla assembler, you'll need to force a cold start of the assembler by typing "NEW" or 1000G as a monitor command; this action will ensure all the internal patches have been installed into DOS as well. The language card version cold starts itself, and requires no intervention.

A cleaner way is to use an EXEC file. The following file will bring up the vanilla version of the assembler:

REM LOAD ASM	
CALL -151	Enter the monitor
BLOAD S-C.ASM.MACRO	Load the Assembler
BLOAD SCM80	Load the patches
4000G	Install them, and
1000G	Start the assembler!

To load the language card patches with an EXEC file, refer to Bob's EXEC file on the top of Page 4 of the May '82 AAL, and replace "3D3G" with the following two lines:

BLOAD SCM80.LC	Load the LC patches
4000G	Install them and cold start!

The character I/O is being vectored through routines at the end of the assembler; for the language card version, these routines are somewhere in \$F4XX. If you decide to issue an "FP" command from that version, you'll find yourself in "Never-Never Land". It's good practice to issue a "PR#n" first (where "n" is the Videoterm's slot number). When you type "INT" to restart the assembler, the special I/O routines will automatically be hooked in.

Decision Systems

Decision Systems
P.O. Box 13006
Denton, TX 76203
817/382-6353

DIS-ASSEMBLER

DSA-DS dis-assembles Apple machine language programs into forms compatible with LISA, S-C ASSEMBLER (3.2 or 4.0), Apple's TOOL-KIT ASSEMBLER and others. DSA-DS dis-assembles instructions or data. Labels are generated for referenced locations within the machine language program.

\$25, Disk, Applesoft (32K, ROM or Language card)

OTHER PRODUCTS

ISAM-DS is an integrated set of Applesoft routines that gives indexed file capabilities to your **BASIC** programs. Retrieve by key, partial key or sequentially. Space from deleted records is automatically reused. Capabilities and performance that match products costing twice as much.

\$50 Disk, Applesoft.

PBASIC-DS is a sophisticated preprocessor for structured **BASIC**. Use advanced logic constructs such as **IF...ELSE...**, **CASE**, **SELECT**, and many more. Develop programs for Integer or Applesoft. Enjoy the power of structured logic at a fraction of the cost of **PASCAL**.

\$35. Disk, Applesoft (48K, ROM or Language Card).

FORM-DS is a complete system for the definition of input and output forms. **FORM-DS** supplies the automatic checking of numeric input for acceptable range of values, automatic formatting of numeric output, and many more features.

\$25 Disk, Applesoft (32K, ROM or Language Card).

UTIL-DS is a set of routines for use with Applesoft to format numeric output, selectively clear variables (Applesoft's **CLEAR** gets everything), improve error handling, and interface machine language with Applesoft programs. Includes a special load routine for placing machine language routines underneath Applesoft programs.

\$25 Disk, Applesoft.

SPEED-DS is a routine to modify the statement linkage in an Applesoft program to speed its execution. Improvements of 5-20% are common. As a bonus, **SPEED-DS** includes machine language routines to speed string handling and reduce the need for garbage clean-up. Author: Lee Meador.

\$15 Disk, Applesoft (32K, ROM or Language Card).

(Add \$4.00 for Foreign Mail)

*Apple II is a registered trademark of the Apple Computer Co.

A Funny Thing Happened on the Way...

Bob thought it would be enlightening to touch on some of the crazy things that went on during the development of these routines. I always marvel at people like Bob, Mike, Bill, and Lee, who have a gift for writing machine language, and can sit down and bang out a line editor in a few hours.

The rest of us aren't quite so fortunate. SCM80 took my three days to write, even though I had done some quick patches on Version 4.0. A couple of good ones popped up during that time, and I'll pass them along.

I was determined to interface the Videoterm using only its terminal functions, avoiding any internal Videoterm ROM routines that would make the interface version-dependent (my card matches neither the descriptions nor the ROM source listings contained in my manual!).

The Videoterm will not move its flashing cursor to a GOTIFY Location unless the cursor is first placed there and then a character is output; under BASIC, you can't just HTAB and VTAB to a position and GET a character -- you have to print a character first (even a null character will do it), in order to move the cursor!

After spending several hours fighting with the Videoterm over who was controlling the input and output cursor locations, I finally decided to designate my own locations for CH and CV (normally at \$24 and \$25) for use by the editing routines.

The other frustration I incurred was doing the case-switching in the replacement RDKEY routine. I was using the language card version, and had carefully checked my code, but the assembler just wouldn't switch case for me. True confession: it took almost fifteen minutes before it dawned on me that the assembler's case flag (at \$D016) was write protected! Hence, the special unprotect subroutine called by the new RDKEY.

One final note concerns the contortions in the replacement COUT and WARM.ENTRY routines (at least I saw these coming!). We need to keep our new RDKEY routine in the DOS input hook to keep things working predictably. The Videoterm, when installed by placing it in the output hook and calling it to output a character, takes over the input hook as well. In addition, we have a replacement COUT routine that is designed to detect and modify control characters for display prior to their output.

In order to avoid arm-wrestling with the Videoterm over who controls the input hook, I used a strange but effective technique. During the installation and patch portion, I install the Videoterm in the designated slot, hook it in, and send a bogus character to make sure it has installed its warm entry I/O locations in DOS (\$AA52-\$AA56 for 48K machines). The code immediately following uses an internal assembler routine to calculate the address of the DOS output hook, regardless of memory size. The contents of the DOS output hook

are then moved into the new COUT routine, immediately following a JMP, and the same COUT routine is forced into the DOS hook, along with the new RDKEY routine. Whenever a character is output, it will first be given to COUT; when COUT has done its work, the character is then passed to the Videoterm's warm entry.

During the installation and patch, the warm start vector within the assembler was modified to point to the NEW.WARM.START routine, which re-installs COUT and RDKEY, keeping everything in sync. A RESET will always restore this condition, no matter what the Videoterm may have in mind!

The S-C Macro Assembler is a wonderful piece of software, and the upgrade is a steal at \$27.50. The only thing that can top it is being able to use it with 80 columns of display!

If you find any errors in my patches, or come up with some new features, contact me at (206) 779-9508.

Supercharge Your APPLE II*



The Axlon RAMDISK™ 320K Memory System for the Apple II and Apple II Plus* provides access speeds never before available. The Axlon memory system is designed to interact with Apple DOS 3.3* and Apple Pascal 1.1* like two standard floppy disk drives while delivering the lightning fast access speeds of RAM memory. This also leaves 32K of RAM for advanced programming techniques. The interface board is slot independent and draws no power from your Apple. The rechargeable battery system built into the unit provides three hours of backup in the event of a power loss. Drop by your local Apple dealer or contact Axlon, Inc. for more information.

* Trademark of Apple Computer, Inc.

* Pascal is a Trademark of U.C.S.D. Regents

- Plug-in compatibility
- 320K bytes of RAM (200NS) memory designed to function like two 35 track floppy disk drives
- Compatible with Apple DOS 3.3 and Apple Pascal 1.1
- Same size as the Apple Disk II* Drive
- Invisible memory refresh - even with the Apple turned off
- Rechargeable battery system built-in to provide 3 hours of auxiliary power
- Slot independent interface board - draws no power from your Apple
- All firmware is in static RAM on the interface board
- Includes software for diagnostic, fast load and copy routines, and business applications

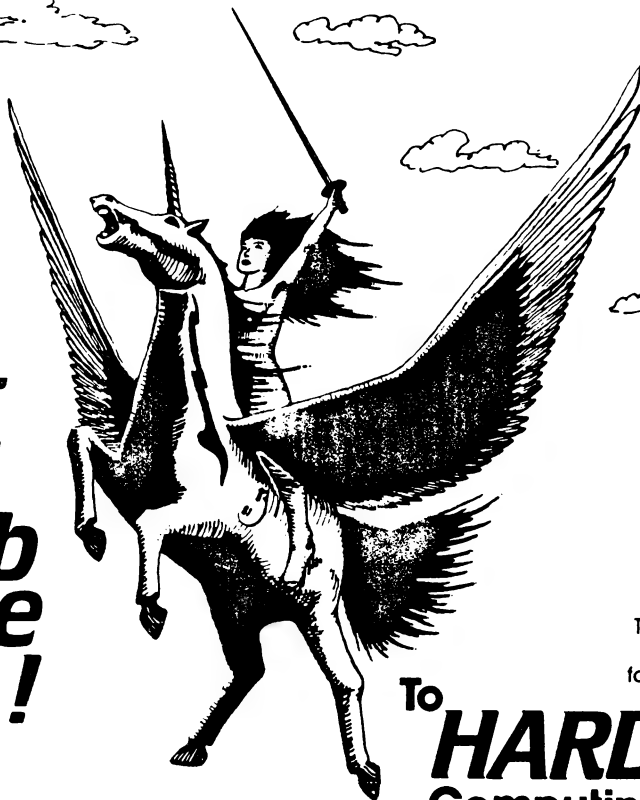


170 N. Wolfe Road,
Sunnyvale, CA 94086
(408) 730-0216

:
:LIST

```
1000      .LIST OFF
1010 *-----
1020 *               SCM80
1030 *   Patches for S-C Macro Assembler V1.0
1040 *       for Vindex Videoterm Card
1050 *
1060 *   Date: 7/10/82
1070 *
1080 *   Don Taylor
1090 *   infoTool corporation
1100 *   Drawer 809, Poulsbo, WA  98370
1110 *
1120 *   To assemble this file:
1130 *
1140 *       1.  Set SLOTNUM to slot number of vindex card
1150 *
1160 *       2.  Set LCVERSION flag for
1170 *           .EQ 1 for Language card version ($D000)
1180 *           .EQ 0 for Standard version ($1000)
1190 *
1200 *       3.  Assemble as usual
1210 *
1220 *       4.  Use VAL LENGTH to get length in hex
1230 *
1240 *       5.  BSAVE SCM80, A$4000, L$LENGTH
1250 *
1260 *-----
1270 *
1280 SLOTNUM      .EQ 3      VINDEX slot
1290 LCVERSION    .EQ 1      SCM80 version
1300 PATCH.AREA  .EQ $3200
1310 LC.PATCH.AREA .EQ $F400
1320 *
1330 *-----
1340 *       .DO LCVERSION
1350 SCM.BASE     .EQ $D000
1360 *       .ELSE
1370 SCM.BASE     .EQ $1000
1380 *       .FIN
1390 *-----
1400 *   Program Constants
1410 *-----
1420 MON.CSW      .EQ $36
1430 MON.KSW      .EQ $38
1440 MON.A1L      .EQ $3C
1450 MON.A2L      .EQ $3E
1460 MON.A4L      .EQ $42
1470 SCM.POINTER .EQ $58
1480 SCM.CURR.CHAR .EQ $61
1490 SCM.ED.BEGLIN .EQ $80
1500 NEW.CH       .EQ $98
1510 NEW.CV       .EQ $99
1520 SCM.WBUF     .EQ $200
1530 DOS.COLD.ENTRY .EQ $3D3
1540 DOS.IOHOOK   .EQ $3EA
1550 FLAGS       .EQ $7F8      VINDEX Flag Byte
1560 KEYBOARD     .EQ $C000
1570 KEYSTROBE    .EQ $C010
1580 SCM.WARM.ENTRY .EQ SCM.BASE+$003
1590 SCM.SHIFT.FLAG .EQ SCM.BASE+$016
1600 SCM.SYM.TABLE .EQ SCM.BASE+$01D
1610 SCM.TEST.DOS  .EQ SCM.BASE+$31E
1620 SCM.RDL.EOL   .EQ SCM.BASE+$35E
1630 SCM.RDL3      .EQ SCM.BASE+$3C3
1640 SCM.ESC.TABLE .EQ SCM.BASE+$467
1650 SCM.ESC.L      .EQ SCM.BASE+$483
1660 SCM.RDKEY.NO.CASE .EQ SCM.BASE+$520
1670 SCM.RDKEY.WITH.CASE .EQ SCM.BASE+$4CA
1680 SCM.SPC       .EQ SCM.BASE+$D92
1690 MON.MOVE     .EQ $FE2C
1700 MON.OUTPORT   .EQ $FE95
1710 MON.COUT      .EQ $FDED
1720 MON.RTS       .EQ $FF58
1730 *-----
```

**S
u
b
s
c
r
i
b
e
!**



The most controversial
computer magazine
for serious Apple-Users.

To
HARDCORE
Computing

Isn't it time that you found out the true story behind the most controversial user-oriented computer magazine around today? Over 64 pages of facts, programming aids and program listings that includes columns on HOW TO: 1. Copy-protect disks, 2. Normalize "unlistable" programs, 3. Use bit-copy programs to make back-ups of the "uncopiables," 4. Write your own adventure-arcade games, 5. Market your software, and 6. Learn all about DOS.

"HARDCORE Computing ... warns pirates about the latest technology that companies are using against them." **TIME**, Feb. 8, 1982

"When some Apple enthusiasts ... heard about the boycott (of bit-copy ads), they concluded that it was nothing but censorship and another example of the magazines ignoring the average Apple user to placate their advertisers. So they started their own publication, HARDCORE Computing ..." **ESQUIRE**, Jan. 1982



HARDCORE COMPUTING

Dept. AL-1
P.O. Box 44549
Tacoma, WA 98444

Subscriptions:

\$20.00 U.S.A.	\$28.50 Canada
\$32.50 Mexico	\$42.00 Others

NAME _____

ADDRESS _____

CITY _____ STATE _____ ZIP _____

```

1740      .OR $H000
1750      START1      EQ *
1760      MOVE CODE
1770      -----
1780      LDA HERE
1790      MOV A,IL
1800      LDA HERE
1810      MOV A,IL+1
1820      LDA HERE
1830      MOV A,IL
1840      LDA HERE
1850      MOV A,IL+1
1860      LDA HERE
1870      MOV A,IL+1
1880      -----
1890      DO LCVERSION
1900      BIT $C083
1910      LDA $C.PATCH.AREA
1920      LDA MON.AIL
1930      LDA /LC.PATCH.AREA
1940      STA MON.AIL+1
1950      ELSE
1960      LDA $M33
1970      LDA $M33.SYM.TABLE
1980      LDA $C.PATCH.AREA
1990      LDA MON.AIL
2000      LDA /PATCH.AREA
2010      STA MON.AIL+1
2020      .FIN
2030      -----
2040      LDY MON.MOVE
2050      INSTALL PATCHES
2060      LDA $EA
2070      STA $CM.BASE+$343
2080      STA $CM.BASE+$344
2090      STA $CM.BASE+$028
2100      STA $CM.BASE+$029
2110      STA $CM.BASE+$02A
2120      LDY $9
2130      STA $CM.BASE+$298,X
2140      DEY
2150      BPL $1
2160      LDY $14
2170      STA $CM.BASE+$4DE,X
2180      DEY
2190      LDY $8
2200      STA $CM.BASE+$B35,X
2210      DEY
2220      BPL $3
2230      -----
2240      LDA $320
2250      STA $CM.BASE+$295
2260      LDA $321
2270      STA $CM.BASE+$296
2280      LDA $322
2290      STA $CM.BASE+$297
2300      LDA $323
2310      STA $CM.BASE+$298
2320      LDA $324
2330      STA $CM.BASE+$299
2340      LDA $325
2350      STA $CM.BASE+$2A7
2360      LDA $326
2370      STA $CM.BASE+$2A8
2380      LDA $327
2390      STA $CM.BASE+$2A9
2300      LDA $328
2410      STA $CM.BASE+$2AA
2420      LDA $329
2430      STA $CM.BASE+$2AB
2440      LDA $330
2450      STA $CM.BASE+$2AC
2460      LDA $331
2470      STA $CM.BASE+$2AD
2480      LDA $332
2490      STA $CM.BASE+$2AE
2500      LDA $333
2510      STA $CM.BASE+$2AF
2520      LDA $334
2530      STA $CM.BASE+$2B0
2540      LDA $335
2550      STA $CM.BASE+$2B1
2560      LDA $336
2570      STA $CM.BASE+$2B2
2580      LDA $337
2590      STA $CM.BASE+$2B3
2600      LDA $338
2610      STA $CM.BASE+$2B4
2620      LDA $339
2630      STA $CM.BASE+$2B5
2640      LDA $340
2650      STA $CM.BASE+$2B6
2660      LDA $341
2670      STA $CM.BASE+$2B7
2680      LDA $342
2690      STA $CM.BASE+$2B8
2700      LDA $343
2710      STA $CM.BASE+$2B9
2720      LDA $344
2730      STA $CM.BASE+$2BA
2740      LDA $345
2750      STA $CM.BASE+$2BB
2760      LDA $346
2770      STA $CM.BASE+$2BC
2780      LDA $347
2790      STA $CM.BASE+$2BD
2800      LDA $348
2810      STA $CM.BASE+$2BE
2820      LDA $349
2830      STA $CM.BASE+$2BF
2840      LDA $350
2850      STA $CM.BASE+$2C0
2860      LDA $351
2870      STA $CM.BASE+$2C1
2880      LDA $352
2890      STA $CM.BASE+$2C2
2900      LDA $353
2910      STA $CM.BASE+$2C3
2920      LDA $354
2930      STA $CM.BASE+$2C4
2940      LDA $355
2950      STA $CM.BASE+$2C5
2960      LDA $356
2970      STA $CM.BASE+$2C6
2980      LDA $357
2990      STA $CM.BASE+$2C7
3000      LDA $358
3010      STA $CM.BASE+$2C8
3020      LDA $359
3030      STA $CM.BASE+$2C9
3040      LDA $360
3050      STA $CM.BASE+$2CA
3060      LDA $361
3070      STA $CM.BASE+$2CB
3080      LDA $362
3090      STA $CM.BASE+$2CC
3100      LDA $363
3110      STA $CM.BASE+$2CD
3120      LDA $364
3130      STA $CM.BASE+$2CE
3140      LDA $365
3150      STA $CM.BASE+$2CF
3160      LDA $366
3170      STA $CM.BASE+$2D0
3180      LDA $367
3190      STA $CM.BASE+$2D1
3200      LDA $368
3210      STA $CM.BASE+$2D2
3220      LDA $369
3230      STA $CM.BASE+$2D3
3240      LDA $370
3250      STA $CM.BASE+$2D4
3260      LDA $371
3270      STA $CM.BASE+$2D5
3280      LDA $372
3290      STA $CM.BASE+$2D6
3300      LDA $373
3310      STA $CM.BASE+$2D7
3320      LDA $374
3330      STA $CM.BASE+$2D8
3340      LDA $375
3350      STA $CM.BASE+$2D9
3360      LDA $376
3370      STA $CM.BASE+$2DA
3380      LDA $377
3390      STA $CM.BASE+$2DB
3400      LDA $378
3410      STA $CM.BASE+$2DC
3420      LDA $379
3430      STA $CM.BASE+$2DD
3440      LDA $380
3450      STA $CM.BASE+$2DE
3460      LDA $381
3470      STA $CM.BASE+$2DF
3480      LDA $382
3490      STA $CM.BASE+$2E0
3500      LDA $383
3510      STA $CM.BASE+$2E1
3520      LDA $384
3530      STA $CM.BASE+$2E2
3540      LDA $385
3550      STA $CM.BASE+$2E3
3560      LDA $386
3570      STA $CM.BASE+$2E4
3580      LDA $387
3590      STA $CM.BASE+$2E5
3600      LDA $388
3610      STA $CM.BASE+$2E6
3620      LDA $389
3630      STA $CM.BASE+$2E7
3640      LDA $390
3650      STA $CM.BASE+$2E8
3660      LDA $391
3670      STA $CM.BASE+$2E9
3680      LDA $392
3690      STA $CM.BASE+$2EA
3700      LDA $393
3710      STA $CM.BASE+$2EB
3720      LDA $394
3730      STA $CM.BASE+$2EC
3740      LDA $395
3750      STA $CM.BASE+$2ED
3760      LDA $396
3770      STA $CM.BASE+$2EE
3780      LDA $397
3790      STA $CM.BASE+$2EF
3800      LDA $398
3810      STA $CM.BASE+$2F0
3820      LDA $399
3830      STA $CM.BASE+$2F1
3840      LDA $400
3850      STA $CM.BASE+$2F2
3860      LDA $401
3870      STA $CM.BASE+$2F3
3880      LDA $402
3890      STA $CM.BASE+$2F4
3900      LDA $403
3910      STA $CM.BASE+$2F5
3920      LDA $404
3930      STA $CM.BASE+$2F6
3940      LDA $405
3950      STA $CM.BASE+$2F7
3960      LDA $406
3970      STA $CM.BASE+$2F8
3980      LDA $407
3990      STA $CM.BASE+$2F9
4000      LDA $408
4010      STA $CM.BASE+$2FA
4020      LDA $409
4030      STA $CM.BASE+$2FB
4040      LDA $410
4050      STA $CM.BASE+$2FC
4060      LDA $411
4070      STA $CM.BASE+$2FD
4080      LDA $412
4090      STA $CM.BASE+$2FE
4100      LDA $413
4110      STA $CM.BASE+$2FF
4120      LDA $414
4130      STA $CM.BASE+$300
4140      LDA $415
4150      STA $CM.BASE+$301
4160      LDA $416
4170      STA $CM.BASE+$302
4180      LDA $417
4190      STA $CM.BASE+$303
4200      LDA $418
4210      STA $CM.BASE+$304
4220      LDA $419
4230      STA $CM.BASE+$305
4240      LDA $420
4250      STA $CM.BASE+$306
4260      LDA $421
4270      STA $CM.BASE+$307
4280      LDA $422
4290      STA $CM.BASE+$308
4300      LDA $423
4310      STA $CM.BASE+$309
4320      LDA $424
4330      STA $CM.BASE+$30A
4340      LDA $425
4350      STA $CM.BASE+$30B
4360      LDA $426
4370      STA $CM.BASE+$30C
4380      LDA $427
4390      STA $CM.BASE+$30D
4400      LDA $428
4410      STA $CM.BASE+$30E
4420      LDA $429
4430      STA $CM.BASE+$30F
4440      LDA $430
4450      STA $CM.BASE+$310
4460      LDA $431
4470      STA $CM.BASE+$311
4480      LDA $432
4490      STA $CM.BASE+$312
4500      LDA $433
4510      STA $CM.BASE+$313
4520      LDA $434
4530      STA $CM.BASE+$314
4540      LDA $435
4550      STA $CM.BASE+$315
4560      LDA $436
4570      STA $CM.BASE+$316
4580      LDA $437
4590      STA $CM.BASE+$317
4600      LDA $438
4610      STA $CM.BASE+$318
4620      LDA $439
4630      STA $CM.BASE+$319
4640      LDA $440
4650      STA $CM.BASE+$31A
4660      LDA $441
4670      STA $CM.BASE+$31B
4680      LDA $442
4690      STA $CM.BASE+$31C
4700      LDA $443
4710      STA $CM.BASE+$31D
4720      LDA $444
4730      STA $CM.BASE+$31E
4740      LDA $445
4750      STA $CM.BASE+$31F
4760      LDA $446
4770      STA $CM.BASE+$320
4780      LDA $447
4790      STA $CM.BASE+$321
4800      LDA $448
4810      STA $CM.BASE+$322
4820      LDA $449
4830      STA $CM.BASE+$323
4840      LDA $450
4850      STA $CM.BASE+$324
4860      LDA $451
4870      STA $CM.BASE+$325
4880      LDA $452
4890      STA $CM.BASE+$326
4900      LDA $453
4910      STA $CM.BASE+$327
4920      LDA $454
4930      STA $CM.BASE+$328
4940      LDA $455
4950      STA $CM.BASE+$329
4960      LDA $456
4970      STA $CM.BASE+$32A
4980      LDA $457
4990      STA $CM.BASE+$32B
5000      LDA $458
5010      STA $CM.BASE+$32C
5020      LDA $459
5030      STA $CM.BASE+$32D
5040      LDA $460
5050      STA $CM.BASE+$32E
5060      LDA $461
5070      STA $CM.BASE+$32F
5080      LDA $462
5090      STA $CM.BASE+$330
5100      LDA $463
5110      STA $CM.BASE+$331
5120      LDA $464
5130      STA $CM.BASE+$332
5140      LDA $465
5150      STA $CM.BASE+$333
5160      LDA $466
5170      STA $CM.BASE+$334
5180      LDA $467
5190      STA $CM.BASE+$335
5200      LDA $468
5210      STA $CM.BASE+$336
5220      LDA $469
5230      STA $CM.BASE+$337
5240      LDA $470
5250      STA $CM.BASE+$338
5260      LDA $471
5270      STA $CM.BASE+$339
5280      LDA $472
5290      STA $CM.BASE+$33A
5300      LDA $473
5310      STA $CM.BASE+$33B
5320      LDA $474
5330      STA $CM.BASE+$33C
5340      LDA $475
5350      STA $CM.BASE+$33D
5360      LDA $476
5370      STA $CM.BASE+$33E
5380      LDA $477
5390      STA $CM.BASE+$33F
5400      LDA $478
5410      STA $CM.BASE+$340
5420      LDA $479
5430      STA $CM.BASE+$341
5440      LDA $480
5450      STA $CM.BASE+$342
5460      LDA $481
5470      STA $CM.BASE+$343
5480      LDA $482
5490      STA $CM.BASE+$344
5500      LDA $483
5510      STA $CM.BASE+$345
5520      LDA $484
5530      STA $CM.BASE+$346
5540      LDA $485
5550      STA $CM.BASE+$347
5560      LDA $486
5570      STA $CM.BASE+$348
5580      LDA $487
5590      STA $CM.BASE+$349
5600      LDA $488
5610      STA $CM.BASE+$34A
5620      LDA $489
5630      STA $CM.BASE+$34B
5640      LDA $490
5650      STA $CM.BASE+$34C
5660      LDA $491
5670      STA $CM.BASE+$34D
5680      LDA $492
5690      STA $CM.BASE+$34E
5700      LDA $493
5710      STA $CM.BASE+$34F
5720      LDA $494
5730      STA $CM.BASE+$350
5740      LDA $495
5750      STA $CM.BASE+$351
5760      LDA $496
5770      STA $CM.BASE+$352
5780      LDA $497
5790      STA $CM.BASE+$353
5800      LDA $498
5810      STA $CM.BASE+$354
5820      LDA $499
5830      STA $CM.BASE+$355
5840      LDA $500
5850      STA $CM.BASE+$356
5860      LDA $501
5870      STA $CM.BASE+$357
5880      LDA $502
5890      STA $CM.BASE+$358
5900      LDA $503
5910      STA $CM.BASE+$359
5920      LDA $504
5930      STA $CM.BASE+$35A
5940      LDA $505
5950      STA $CM.BASE+$35B
5960      LDA $506
5970      STA $CM.BASE+$35C
5980      LDA $507
5990      STA $CM.BASE+$35D
6000      LDA $508
6010      STA $CM.BASE+$35E
6020      LDA $509
6030      STA $CM.BASE+$35F
6040      LDA $510
6050      STA $CM.BASE+$360
6060      LDA $511
6070      STA $CM.BASE+$361
6080      LDA $512
6090      STA $CM.BASE+$362
6100      LDA $513
6110      STA $CM.BASE+$363
6120      LDA $514
6130      STA $CM.BASE+$364
6140      LDA $515
6150      STA $CM.BASE+$365
6160      LDA $516
6170      STA $CM.BASE+$366
6180      LDA $517
6190      STA $CM.BASE+$367
6200      LDA $518
6210      STA $CM.BASE+$368
6220      LDA $519
6230      STA $CM.BASE+$369
6240      LDA $520
6250      STA $CM.BASE+$36A
6260      LDA $521
6270      STA $CM.BASE+$36B
6280      LDA $522
6290      STA $CM.BASE+$36C
6300      LDA $523
6310      STA $CM.BASE+$36D
6320      LDA $524
6330      STA $CM.BASE+$36E
6340      LDA $525
6350      STA $CM.BASE+$36F
6360      LDA $526
6370      STA $CM.BASE+$370
6380      LDA $527
6390      STA $CM.BASE+$371
6400      LDA $528
6410      STA $CM.BASE+$372
6420      LDA $529
6430      STA $CM.BASE+$373
6440      LDA $530
6450      STA $CM.BASE+$374
6460      LDA $531
6470      STA $CM.BASE+$375
6480      LDA $532
6490      STA $CM.BASE+$376
6500      LDA $533
6510      STA $CM.BASE+$377
6520      LDA $534
6530      STA $CM.BASE+$378
6540      LDA $535
6550      STA $CM.BASE+$379
6560      LDA $536
6570      STA $CM.BASE+$37A
6580      LDA $537
6590      STA $CM.BASE+$37B
6600      LDA $538
6610      STA $CM.BASE+$37C
6620      LDA $539
6630      STA $CM.BASE+$37D
6640      LDA $540
6650      STA $CM.BASE+$37E
6660      LDA $541
6670      STA $CM.BASE+$37F
6680      LDA $542
6690      STA $CM.BASE+$380
6700      LDA $543
6710      STA $CM.BASE+$381
6720      LDA $544
6730      STA $CM.BASE+$382
6740      LDA $545
6750      STA $CM.BASE+$383
6760      LDA $546
6770      STA $CM.BASE+$384
6780      LDA $547
6790      STA $CM.BASE+$385
6800      LDA $548
6810      STA $CM.BASE+$386
6820      LDA $549
6830      STA $CM.BASE+$387
6840      LDA $550
6850      STA $CM.BASE+$388
6860      LDA $551
6870      STA $CM.BASE+$389
6880      LDA $552
6890      STA $CM.BASE+$38A
6900      LDA $553
6910      STA $CM.BASE+$38B
6920      LDA $554
6930      STA $CM.BASE+$38C
6940      LDA $555
6950      STA $CM.BASE+$38D
6960      LDA $556
6970      STA $CM.BASE+$38E
6980      LDA $557
6990      STA $CM.BASE+$38F
7000      LDA $558
7010      STA $CM.BASE+$390
7020      LDA $559
7030      STA $CM.BASE+$391
7040      LDA $560
7050      STA $CM.BASE+$392
7060      LDA $561
7070      STA $CM.BASE+$393
7080      LDA $562
7090      STA $CM.BASE+$394
7100      LDA $563
7110      STA $CM.BASE+$395
7120      LDA $564
7130      STA $CM.BASE+$396
7140      LDA $565
7150      STA $CM.BASE+$397
7160      LDA $566
7170      STA $CM.BASE+$398
7180      LDA $567
7190      STA $CM.BASE+$399
7200      LDA $568
7210      STA $CM.BASE+$39A
7220      LDA $569
7230      STA $CM.BASE+$39B
7240      LDA $570
7250      STA $CM.BASE+$39C
7260      LDA $571
7270      STA $CM.BASE+$39D
7280      LDA $572
7290      STA $CM.BASE+$39E
7300      LDA $573
7310      STA $CM.BASE+$39F
7320      LDA $574
7330      STA $CM.BASE+$3A0
7340      LDA $575
7350      STA $CM.BASE+$3A1
7360      LDA $576
7370      STA $CM.BASE+$3A2
7380      LDA $577
7390      STA $CM.BASE+$3A3
7400      LDA $578
7410      STA $CM.BASE+$3A4
7420      LDA $579
7430      STA $CM.BASE+$3A5
7440      LDA $580
7450      STA $CM.BASE+$3A6
7460      LDA $581
7470      STA $CM.BASE+$3A7
7480      LDA $582
7490      STA $CM.BASE+$3A8
7500      LDA $583
7510      STA $CM.BASE+$3A9
7520      LDA $584
7530      STA $CM.BASE+$3AA
7540      LDA $585
7550      STA $CM.BASE+$3AB
7560      LDA $586
7570      STA $CM.BASE+$3AC
7580      LDA $587
7590      STA $CM.BASE+$3AD
7600      LDA $588
7610      STA $CM.BASE+$3AE
7620      LDA $589
7630      STA $CM.BASE+$3AF
7640      LDA $590
7650      STA $CM.BASE+$3B0
7660      LDA $591
7670      STA $CM.BASE+$3B1
7680      LDA $592
7690      STA $CM.BASE+$3B2
7700      LDA $593
7710      STA $CM.BASE+$3B3
7720      LDA $594
7730      STA $CM.BASE+$3B4
7740      LDA $595
7750      STA $CM.BASE+$3B5
7760      LDA $596
7770      STA $CM.BASE+$3B6
7780      LDA $597
7790      STA $CM.BASE+$3B7
7800      LDA $598
7810      STA $CM.BASE+$3B8
7820      LDA $599
7830      STA $CM.BASE+$3B9
7840      LDA $600
7850      STA $CM.BASE+$3BA
7860      LDA $601
7870      STA $CM.BASE+$3BB
7880      LDA $602
7890      STA $CM.BASE+$3BC
7900      LDA $603
7910      STA $CM.BASE+$3BD
7920      LDA $604
7930      STA $CM.BASE+$3BE
7940      LDA $605
7950      STA $CM.BASE+$3BF
7960      LDA $606
7970      STA $CM.BASE+$3C0
7980      LDA $607
7990      STA $CM.BASE+$3C1
8000      LDA $608
8010      STA $CM.BASE+$3C2
8020      LDA $609
8030      STA $CM.BASE+$3C3
8040      LDA $610
8050      STA $CM.BASE+$3C4
8060      LDA $611
8070      STA $CM.BASE+$3C5
8080      LDA $612
8090      STA $CM.BASE+$3C6
8100      LDA $613
8110      STA $CM.BASE+$3C7
8120      LDA $614
8130      STA $CM.BASE+$3C8
8140      LDA $615
8150      STA $CM.BASE+$3C9
8160      LDA $616
8170      STA $CM.BASE+$3CA
8180      LDA $617
8190      STA $CM.BASE+$3CB
8200      LDA $618
8210      STA $CM.BASE+$3CC
8220      LDA $619
8230      STA $CM.BASE+$3CD
8240      LDA $620
8250      STA $CM.BASE+$3CE
8260      LDA $621
8270      STA $CM.BASE+$3CF
8280      LDA $622
8290      STA $CM.BASE+$3D0
8300      LDA $623
8310      STA $CM.BASE+$3D1
8320      LDA $624
8330      STA $CM.BASE+$3D2
8340      LDA $625
8350      STA $CM.BASE+$3D3
8360      LDA $626
8370      STA $CM.BASE+$3D4
8380      LDA $627
8390      STA $CM.BASE+$3D5
8400      LDA $628
8410      STA $CM.BASE+$3D6
8420      LDA $629
8430      STA $CM.BASE+$3D7
8440      LDA $630
8450      STA $CM.BASE+$3D8
8460      LDA $631
8470      STA $CM.BASE+$3D9
8480      LDA $632
8490      STA $CM.BASE+$3DA
8500      LDA $633
8510      STA $CM.BASE+$3DB
8520      LDA $634
8530      STA $CM.BASE+$3DC
8540      LDA $635
8550      STA $CM.BASE+$3DD
8560      LDA $636
8570      STA $CM.BASE+$3DE
8580      LDA $637
8590      STA $CM.BASE+$3DF
8600      LDA $638
8610      STA $CM.BASE+$3E0
8620      LDA $639
8630      STA $CM.BASE+$3E1
8640      LDA $640
8650      STA $CM.BASE+$3E2
8660      LDA $641
8670      STA $CM.BASE+$3E3
8680      LDA $642
8690      STA $CM.BASE+$3E4
8700      LDA $643
8710      STA $CM.BASE+$3E5
8720      LDA $644
8730      STA $CM.BASE+$3E6
8740      LDA $645
8750      STA $CM.BASE+$3E7
8760      LDA $646
8770      STA $CM.BASE+$3E8
8780      LDA $647
8790      STA $CM.BASE+$3E9
8800      LDA $648
8810      STA $CM.BASE+$3EA
8820      LDA $649
8830      STA $CM.BASE+$3EB
8840      LDA $650
8850      STA $CM.BASE+$3EC
8860      LDA $651
8870      STA $CM.BASE+$3ED
8880      LDA $652
8890      STA $CM.BASE+$3EE
8900      LDA $653
8910      STA $CM.BASE+$3EF
8920      LDA $654
8930      STA $CM.BASE+$3F0
8940      LDA $655
8950      STA $CM.BASE+$3F1
8960      LDA $656
8970      STA $CM.BASE+$3F2
8980      LDA $657
8990      STA $CM.BASE+$3F3
9000      LDA $658
9010      STA $CM.BASE+$3F4
9020      LDA $659
9030      STA $CM.BASE+$3F5
9040      LDA $660
9050      STA $CM.BASE+$3F6
9060      LDA $661
9070      STA $CM.BASE+$3F7
9080      LDA $662
9090      STA $CM.BASE+$3F8
9100      LDA $663
9110      STA $CM.BASE+$3F9
9120      LDA $664
9130      STA $CM.BASE+$3FA
9140      LDA $665
9150      STA $CM.BASE+$3FB
9160      LDA $666
9170      STA $CM.BASE+$3FC
9180      LDA $667
9190      STA $CM.BASE+$3FD
9200      LDA $668
9210      STA $CM.BASE+$3FE
9220      LDA $669
9230      STA $CM.BASE+$3FF
9240      LDA $670
9250      STA $CM.BASE+$400
9260      LDA $671
9270      STA $CM.BASE+$401
9280      LDA $672
9290      STA $CM.BASE+$402
9300      LDA $673
9310      STA $CM.BASE+$403
9320      LDA $674
9330      STA $CM.BASE+$404
9340      LDA $675
9350      STA $CM.BASE+$405
9360      LDA $676
9370      STA $CM.BASE+$406
9380      LDA $677
9390      STA $CM.BASE+$407
9400      LDA $678
9410      STA $CM.BASE+$408
9420      LDA $679
9430      STA $CM.BASE+$409
9440      LDA $680
9450      STA $CM.BASE+$40A
9460      LDA $681
9470      STA $CM.BASE+$40B
9480      LDA $682
9490      STA $CM.BASE+$40C
9500      LDA $683
9510      STA $CM.BASE+$40D
9520      LDA $684
9530      STA $CM.BASE+$40E
9540      LDA $685
9550      STA $CM.BASE+$40F
9560      LDA $686
9570      STA $CM.BASE+$410
9580      LDA $687
9590      STA $CM.BASE+$411
9600      LDA $688
9610      STA $CM.BASE+$412
9620      LDA $689
9630      STA $CM.BASE+$413
9640      LDA $690
9650      STA $CM.BASE+$414
9660      LDA $691
9670      STA $CM.BASE+$415
9680      LDA $692
9690      STA $CM.BASE+$416
9700      LDA $693
9710      STA $CM.BASE+$417
9720      LDA $694
9730      STA $CM.BASE+$418
9740      LDA $695
9750      STA $CM.BASE+$419
9760      LDA $696
9770      STA $CM.BASE+$41A
9780      LDA $697
9790      STA $CM.BASE+$41B
9800      LDA $698
9810      STA $CM.BASE+$41C
9820      LDA $699
9830      STA $CM.BASE+$41D
9840      LDA $700
9850      STA $CM.BASE+$41E
9860      LDA $701
9870      STA $CM.BASE+$41F
9880      LDA $702
9890      STA $CM.BASE+$420
9900      LDA $703
9910      STA $CM.BASE+$421
9920      LDA $704
9930      STA $CM.BASE+$422
9940      LDA $705
9950      STA $CM.BASE+$423
9960      LDA $706
9970      STA $CM.BASE+$424
9980      LDA $707
9990      STA $CM.BASE+$425
10000      LDA $708
10010      STA $CM.BASE+$426
10020      LDA $709
10030      STA $CM.BASE+$427
10040      LDA $710
10050      STA $CM.BASE+$428
10060      LDA $711
10070      STA $CM.BASE+$429
10080      LDA $712
10090      STA $CM.BASE+$42A
10100      LDA $713
10110      STA $CM.BASE+$42B
10120      LDA $714
10130      STA $CM.BASE+$42C
10140      LDA $715
10150      STA $CM.BASE+$42D
10160      LDA $716
10170      STA $CM.BASE+$42E
10180      LDA $717
10190      STA $CM.BASE+$42F
10200      LDA $718
10210      STA $CM.BASE+$430
10220      LDA $719
10230      STA $CM.BASE+$431
10240      LDA $720
10250      STA $CM.BASE+$432
10260      LDA $721
10270      STA $CM.BASE+$433
10280      LDA $722
10290      STA $CM.BASE+$434
10300      LDA $723
10310      STA $CM.BASE+$435
10320      LDA $724
10330      STA $CM.BASE+$436
10340      LDA $725
10350      STA $CM.BASE+$437
10360      LDA $726
10370      STA $CM.BASE+$438
10380      LDA $727
10390      STA $CM.BASE+$439
10400      LDA $728
10410      STA $CM.BASE+$43A
10420      LDA $729
10430      STA $CM.BASE+$43B
10440      LDA $730
10450      STA $CM.BASE+$43C
10460      LDA $731
10470      STA $CM.BASE+$43D
10480      LDA $732
10490      STA $CM.BASE+$43E
10500      LDA $733
10510      STA $CM.BASE+$43F
10520      LDA $734
10530      STA $CM.BASE+$440
10540      LDA $735
10550      STA $CM.BASE+$441
10560      LDA $736
10570      STA $CM.BASE+$442
10580      LDA $737
10590      STA $CM.BASE+$443
10600      LDA $738
10610      STA $CM.BASE+$444
10620      LDA $739
10630      STA $CM.BASE+$445
10640      LDA $740
10650      STA $CM.BASE+$446
10660      LDA $741
10670      STA $CM.BASE+$447
10680      LDA $742
10690      STA $CM.BASE+$448
10700      LDA $743
10710      STA $CM.BASE+$449
10720      LDA $744
10730      STA $CM.BASE+$44A
10740      LDA $745
10750      STA $CM.BASE+$44B
10760      LDA $746
10770      STA $CM.BASE+$44C
10780      LDA $747
10790      STA $CM.BASE+$44D
10800      LDA $748
10810      STA $CM.BASE+$44E
10820      LDA $749
10830      STA $CM.BASE+$44F
10840      LDA $750
10850      STA $CM.BASE+$450
10860      LDA $751
10870      STA $CM.BASE+$451
10880      LDA $752
10890      STA $CM.BASE+$452
10900      LDA $753
10910      STA $CM.BASE+$453
10920      LDA $754
10930      STA $CM.BASE+$454
10940      LDA $755
10950      STA $CM.BASE+$455
10960      LDA $756
10970      STA $CM.BASE+$456
10980      LDA $757
10990      STA $CM.BASE+$457
11000      LDA $758
11010      STA $CM.BASE+$458
11020      LDA $759
11030      STA $CM.BASE+$459
11040      LDA $760
11050      STA $CM.BASE+$45A
11060      LDA $761
11070      STA $CM.BASE+$45B
11080      LDA $762
11090      STA $CM.BASE+$45C
11100      LDA $763
11110      STA $CM.BASE+$45D
11120      LDA $764
11130      STA $CM.BASE+$45E
11140      LDA $765
11150      STA $CM.BASE+$45F
11160      LDA $766
11170      STA $CM.BASE+$460
11180      LDA $767
11190      STA $CM.BASE+$461
11200      LDA $768
11210      STA $CM.BASE+$462
11220      LDA $769
11230      STA $CM.BASE+$463
11240      LDA $770
11250      STA $CM.BASE+$464
11260      LDA $771
11270      STA $CM.BASE+$465
11280      LDA $772
11290      STA $CM.BASE+$466
11300      LDA $773
11310      STA $CM.BASE+$467
11320      LDA $774
11330      STA $CM.BASE+$468
11340      LDA $775
11350      STA $CM.BASE+$469
11360      LDA $776
11370      STA $CM.BASE+$46A
11380      LDA $777
11390      STA $CM.BASE+$46B
11400      LDA $778
11410      STA $CM.BASE+$46C
11420      LDA $779
11430      STA $CM.BASE+$46D
11440      LDA $780
11450      STA $CM.BASE+$46E
11460      LDA $781
11470      STA $CM.BASE+$46F
11480      LDA $782
11490      STA $CM.BASE+$470
11500      LDA $783
11510      STA $CM.BASE+$471
11520      LDA $784
11530      STA $CM.BASE+$472
11540      LDA $785
11550      STA $CM.BASE+$473
11560      LDA $786
11570      STA $CM.BASE+$474
11580      LDA $787
11590      STA $CM.BASE+$475
11600      LDA $788
11610      STA $CM.BASE+$476
11620      LDA $789

```


write now

Southwestern Data Systems, an industry pioneer in innovative software for the Apple II, is always looking for authors. There are no limitations on the size or type of software you can submit — utilities, communication, business, education, or games — the only requirement is that it must meet the quality standards which typify all SDS products. When you join the SDS team, you get the benefits of a professional support staff experienced in providing all you need to get your program to market. Here are some of the ways we help you:

- TECHNICAL PROGRAMMING ASSISTANCE
- UNIQUE COPY PROTECTION W/LIMITED BACKUPS
- SUCCESSFUL MARKETING STRATEGIES
- ASSISTANCE IN WRITING THE MANUAL
- PROFESSIONAL PRODUCT ARTWORK
- QUALITY ADVERTISING
- SUPERIOR PACKAGING
- NATIONAL DISTRIBUTION
- HIGHEST ROYALTIES PAID MONTHLY
- CUSTOMER SERVICE SUPPORT

This is the opportunity you have been waiting for, a chance to market your program with the finest publisher in the software industry. Let Southwestern Data Systems' reputation and proven track record for success go to work for you. If you think you have what we want — a unique and distinctive software package — please call or write us today!

SDS southwestern data systems

P.O. BOX 582 SANTEE, CA 92071 (714) 562-3670

```

4230 UNPROTECT.LC.RAM      Enable Bank 2
4231 BIT $C083
4232 RTS
4233
4234
4235
4236
4237
4238
4239
4240
4241
4242
4243
4244
4245
4246
4247
4248
4249
4250
4251
4252
4253
4254
4255
4256
4257
4258
4259
4260
4261
4262
4263
4264
4265
4266
4267
4268
4269
4270
4271
4272
4273
4274
4275
4276
4277
4278
4279
4280
4281
4282
4283
4284
4285
4286
4287
4288
4289
4290
4291
4292
4293
4294
4295
4296
4297
4298
4299
4300
4301
4302
4303
4304
4305
4306
4307
4308
4309
4310
4311
4312
4313
4314
4315
4316
4317
4318
4319
4320
4321
4322
4323
4324
4325
4326
4327
4328
4329
4330
4331
4332
4333
4334
4335
4336
4337
4338
4339
4340
4341
4342
4343
4344
4345
4346
4347
4348
4349
4350
4351
4352
4353
4354
4355
4356
4357
4358
4359
4360
4361
4362
4363
4364
4365
4366
4367
4368
4369
4370
4371
4372
4373
4374
4375
4376
4377
4378
4379
4380
4381
4382
4383
4384
4385
4386
4387
4388
4389
4390
4391
4392
4393
4394
4395
4396
4397
4398
4399
4400
4401
4402
4403
4404
4405
4406
4407
4408
4409
4410
4411
4412
4413
4414
4415
4416
4417
4418
4419
4420
4421
4422
4423
4424
4425
4426
4427
4428
4429
4430
4431
4432
4433
4434
4435
4436
4437
4438
4439
4440
4441
4442
4443
4444
4445
4446
4447
4448
4449
4450
4451
4452
4453
4454
4455
4456
4457
4458
4459
4460
4461
4462
4463
4464
4465
4466
4467
4468
4469
4470
4471
4472
4473
4474
4475
4476
4477
4478
4479
4480
4481
4482
4483
4484
4485
4486
4487
4488
4489
4490
4491
4492
4493
4494
4495
4496
4497
4498
4499
4500
4501
4502
4503
4504
4505
4506
4507
4508
4509
4510
4511
4512
4513
4514
4515
4516
4517
4518
4519
4520
4521
4522
4523
4524
4525
4526
4527
4528
4529
4530
4531
4532
4533
4534
4535
4536
4537
4538
4539
4540
4541
4542
4543
4544
4545
4546
4547
4548
4549
4550
4551
4552
4553
4554
4555
4556
4557
4558
4559
4560
4561
4562
4563
4564
4565
4566
4567
4568
4569
4570
4571
4572
4573
4574
4575
4576
4577
4578
4579
4580
4581
4582
4583
4584
4585
4586
4587
4588
4589
4590
4591
4592
4593
4594
4595
4596
4597
4598
4599
4600
4601
4602
4603
4604
4605
4606
4607
4608
4609
4610
4611
4612
4613
4614
4615
4616
4617
4618
4619
4620
4621
4622
4623
4624
4625
4626
4627
4628
4629
4630
4631
4632
4633
4634
4635
4636
4637
4638
4639
4640
4641
4642
4643
4644
4645
4646
4647
4648
4649
4650
4651
4652
4653
4654
4655
4656
4657
4658
4659
4660
4661
4662
4663
4664
4665
4666
4667
4668
4669
4670
4671
4672
4673
4674
4675
4676
4677
4678
4679
4680
4681
4682
4683
4684
4685
4686
4687
4688
4689
4690
4691
4692
4693
4694
4695
4696
4697
4698
4699
4700
4701
4702
4703
4704
4705
4706
4707
4708
4709
4710
4711
4712
4713
4714
4715
4716
4717
4718
4719
4720
4721
4722
4723
4724
4725
4726
4727
4728
4729
4730
4731
4732
4733
4734
4735
4736
4737
4738
4739
4740
4741
4742
4743
4744
4745
4746
4747
4748
4749
4750
4751
4752
4753
4754
4755
4756
4757
4758
4759
4760
4761
4762
4763
4764
4765
4766
4767
4768
4769
4770
4771
4772
4773
4774
4775
4776
4777
4778
4779
4780
4781
4782
4783
4784
4785
4786
4787
4788
4789
4790
4791
4792
4793
4794
4795
4796
4797
4798
4799
4800
4801
4802
4803
4804
4805
4806
4807
4808
4809
4810
4811
4812
4813
4814
4815
4816
4817
4818
4819
4820
4821
4822
4823
4824
4825
4826
4827
4828
4829
4830
4831
4832
4833
4834
4835
4836
4837
4838
4839
4840
4841
4842
4843
4844
4845
4846
4847
4848
4849
4850
4851
4852
4853
4854
4855
4856
4857
4858
4859
4860
4861
4862
4863
4864
4865
4866
4867
4868
4869
4870
4871
4872
4873
4874
4875
4876
4877
4878
4879
4880
4881
4882
4883
4884
4885
4886
4887
4888
4889
4890
4891
4892
4893
4894
4895
4896
4897
4898
4899
4900
4901
4902
4903
4904
4905
4906
4907
4908
4909
4910
4911
4912
4913
4914
4915
4916
4917
4918
4919
4920
4921
4922
4923
4924
4925
4926
4927
4928
4929
4930
4931
4932
4933
4934
4935
4936
4937
4938
4939
4940
4941
4942
4943
4944
4945
4946
4947
4948
4949
4950
4951
4952
4953
4954
4955
4956
4957
4958
4959
4960
4961
4962
4963
4964
4965
4966
4967
4968
4969
4970
4971
4972
4973
4974
4975
4976
4977
4978
4979
4980
4981
4982
4983
4984
4985
4986
4987
4988
4989
4990
4991
4992
4993
4994
4995
4996
4997
4998
4999
5000
5001
5002
5003
5004
5005
5006
5007
5008
5009
5010
5011
5012
5013
5014
5015
5016
5017
5018
5019
5020
5021
5022
5023
5024
5025
5026
5027
5028
5029
5030
5031
5032
5033
5034
5035
5036
5037
5038
5039
5040
5041
5042
5043
5044
5045
5046
5047
5048
5049
5050
5051
5052
5053
5054
5055
5056
5057
5058
5059
5060
5061
5062
5063
5064
5065
5066
5067
5068
5069
5070
5071
5072
5073
5074
5075
5076
5077
5078
5079
5080
5081
5082
5083
5084
5085
5086
5087
5088
5089
5090
5091
5092
5093
5094
5095
5096
5097
5098
5099
5100
5101
5102
5103
5104
5105
5106
5107
5108
5109
5110
5111
5112
5113
5114
5115
5116
5117
5118
5119
5120
5121
5122
5123
5124
5125
5126
5127
5128
5129
5130
5131
5132
5133
5134
5135
5136
5137
5138
5139
5140
5141
5142
5143
5144
5145
5146
5147
5148
5149
5150
5151
5152
5153
5154
5155
5156
5157
5158
5159
5160
5161
5162
5163
5164
5165
5166
5167
5168
5169
5170
5171
5172
5173
5174
5175
5176
5177
5178
5179
5180
5181
5182
5183
5184
5185
5186
5187
5188
5189
5190
5191
5192
5193
5194
5195
5196
5197
5198
5199
5200
5201
5202
5203
5204
5205
5206
5207
5208
5209
5210
5211
5212
5213
5214
5215
5216
5217
5218
5219
5220
5221
5222
5223
5224
5225
5226
5227
5228
5229
5230
5231
5232
5233
5234
5235
5236
5237
5238
5239
5240
5241
5242
5243
5244
5245
5246
5247
5248
5249
5250
5251
5252
5253
5254
5255
5256
5257
5258
5259
5260
5261
5262
5263
5264
5265
5266
5267
5268
5269
5270
5271
5272
5273
5274
5275
5276
5277
5278
5279
5280
5281
5282
5283
5284
5285
5286
5287
5288
5289
5290
5291
5292
5293
5294
5295
5296
5297
5298
5299
5300
5301
5302
5303
5304
5305
5306
5307
5308
5309
5310
5311
5312
5313
5314
5315
5316
5317
5318
5319
5320
5321
5322
5323
5324
5325
5326
5327
5328
5329
5330
5331
5332
5333
5334
5335
5336
5337
5338
5339
5340
5341
5342
5343
5344
5345
5346
5347
5348
5349
5350
5351
5352
5353
5354
5355
5356
5357
5358
5359
5360
5361
5362
5363
5364
5365
5366
5367
5368
5369
5370
5371
5372
5373
5374
5375
5376
5377
5378
5379
5380
5381
5382
5383
5384
5385
5386
5387
5388
5389
5390
5391
5392
5393
5394
5395
5396
5397
5398
5399
5400
5401
5402
5403
5404
5405
5406
5407
5408
5409
5410
5411
5412
5413
5414
5415
5416
5417
5418
5419
5420
5421
5422
5423
5424
5425
5426
5427
5428
5429
5430
5431
5432
5433
5434
5435
5436
5437
5438
5439
5440
5441
5442
5443
5444
5445
5446
5447
5448
5449
5450
5451
5452
5453
5454
5455
5456
5457
5458
5459
5460
5461
5462
5463
5464
5465
5466
5467
5468
5469
5470
5471
5472
5473
5474
5475
5476
5477
5478
5479
5480
5481
5482
5483
5484
5485
5486
5487
5488
5489
5490
5491
5492
5493
5494
5495
5496
5497
5498
5499
5500
5501
5502
5503
5504
5505
5506
5507
5508
5509
5510
5511
5512
5513
5514
5515
5516
5517
5518
5519
5520
5521
5522
5523
5524
5525
5526
5527
5528
5529
5530
5531
5532
5533
5534
5535
5536
5537
5538
5539
5540
5541
5542
5543
5544
5545
5546
5547
5548
5549
5550
5551
5552
5553
5554
5555
5556
5557
5558
5559
5560
5561
5562
5563
5564
5565
5566
5567
5568
5569
5570
5571
5572
5573
5574
5575
5576
5577
5578
5579
5580
5581
5582
5583
5584
5585
5586
5587
5588
5589
5590
5591
5592
5593
5594
5595
5596
5597
5598
5599
5600
5601
5602
5603
5604
5605
5606
5607
5608
5609
5610
5611
5612
5613
5614
5615
5616
5617
5618
5619
5620
5621
5622
5623
5624
5625
5626
5627
5628
5629
5630
5631
5632
5633
5634
5635
5636
5637
5638
5639
5640
5641
5642
5643
5644
5645
5646
5647
5648
5649
5650
5651
5652
5653
5654
5655
5656
5657
5658
5659
5660
5661
5662
5663
5664
5665
5666
5667
5668
5669
5670
5671
5672
5673
5674
5675
5676
5677
5678
5679
5680
5681
5682
5683
5684
5685
5686
5687
5688
5689
5690
5691
5692
5693
5694
5695
5696
5697
5698
5699
5700
5701
5702
5703
5704
5705
5706
5707
5708
5709
5710
5711
5712
5713
5714
5715
5716
5717
5718
5719
5720
5721
5722
5723
5724
5725
5726
5727
5728
5729
5730
5731
5732
5733
5734
5735
5736
5737
5738
5739
5740
5741
5742
5743
5744
5745
5746
5747
5748
5749
5750
5751
5752
5753
5754
5755
5756
5757
5758
5759
5760
5761
5762
5763
5764
5765
5766
5767
5768
5769
5770
5771
5772
5773
5774
5775
5776
5777
5778
5779
5780
5781
5782
5783
5784
5785
5786
5787
5788
5789
5790
5791
5792
5793
5794
5795
5796
5797
5798
5799
5800
5801
5802
5803
5804
5805
5806
5807
5808
5809
5810
5811
5812
5813
5814
5815
5816
5817
5818
5819
5820
5821
5822
5823
5824
5825
5826
5827
5828
5829
5830
5831
5832
5833
5834
5835
5836
5837
5838
5839
5840
5841
5842
5843
5844
5845
5846
5847
5848
5849
5850
5851
5852
5853
5854
5855
5856
5857
5858
5859
5860
5861
5862
5863
5864
5865
5866
5867
5868
5869
5870
5871
5872
5873
5874
5875
5876
5877
5878
5879
5880
5881
5882
5883
5884
5885
5886
5887
5888
5889
5890
5891
5892
5893
5894
5895
5896
5897
5898
5899
5900
5901
5902
5903
5904
5905
5906
5907
5908
5909
5910
5911
5912
5913
5914
5915
5916
5917
5918
5919
5920
5921
5922
5923
5924
5925
5926
5927
5928
5929
5930
5931
5932
5933
5934
5935
5936
5937
5938
5939
5940
5941
5942
5943
5944
5945
5946
5947
5948
5949
5950
5951
5952
5953
5954
5955
5956
5957
5958
5959
5960
5961
5962
5963
5964
5965
5966
5967
5968
5969
5970
5971
5972
5973
5974
5975
5976
5977
5978
5979
5980
5981
5982
5983
5984
5985
5986
5987
5988
5989
5990
5991
5992
5993
5994
5995
5996
5997
5998
5999
6000
6001
6002
6003
6004
6005
6006
6007
6008
6009
6010
6011
6012
6013
6014
6015
6016
6017
6018
6019
6020
6021
6022
6023
6024
6025
6026
6027
6028
6029
6030
6031
6032
6033
6034
6035
6036
6037
6038
6039
6040
6041
6042
6043
6044
6045
6046
6047
6048
6049
6050
6051
6052
6053
6054
6055
6056
6057
6058
6059
6060
6061
6062
6063
6064
6065
6066
6067
6068
6069
6070
6071
6072
6073
6074
6075
6076
6077
6078
6079
6080
6081
6082
6083
6084
6085
6086
6087
6088
6089
6090
6091
6092
6093
6094
6095
6096
6097
6098
6099
6100
6101
6102
6103
6104
6105
6106
6107
6108
6109
6110
6111
6112
6113
6114
6115
6116
6117
6118
6119
6120
6121
6122
6123
6124
6125
6126
6127
6128
6129
6130
6131
6132
6133
6134
6135
6136
6137
6138
6139
6140
6141
6142
6143
6144
6145
6146
6147
6148
6149
6150
6151
6152
6153
6154
6155
6156
6157
6158
6159
6160
6161
6162
6163
6164
6165
6166
6167
6168
6169
6170
6171
6172
6173
6174
6175
6176
6177
6178
6179
6180
6181
6182
6183
6184
6185
6186
6187
6188
6189
6190
6191
6192
6193
6194
6195
6196
6197
6198
6199
6200
6201
6202
6203
6204
6205
6206
6207
6208
6209
6210
6211
6212
6213
6214
6215
6216
6217
6218
6219
6220
6221
6222
6223
6224
6225
6226
6227
6228
6229
6230
6231
6232
6233
6234
6235
6236
6237
6238
6239
6240
6241
6242
6243
6244
6245
6246
6247
6248
6249
6250
6251
6252
6253
6254
6255
6256
6257
6258
6259
6260
6261
6262
6263
6264
6265
6266
6267
6268
6269
6270
6271
6272
6273
6274
6275
6276
6277
6278
6279
6280
6281
6282
6283
6284
6285
6286
6287
6288
6289
6290
6291
6292
6293
6294
6295
6296
6297
6298
6299
6300
6301
6302
6303
6304
6305
6306
6307
6308
6309
6310
6311
6312
6313
6314
6315
6316
6317
6318
6319
6320
6321
6322
6323
6324
6325
6326
6327
6328
6329
6330
6331
6332
6333
6334
6335
6336
6337
6338
6339
6340
6341
6342
6343
6344
6345
6346
6347
6348
6349
6350
6351
6352
6353
6354
6355
6356
6357
6358
6359
6360
6361
6362
6363
6364
6365
6366
6367
6368
6369
6370
6371
6372
6373
6374
6375
6376
6377
6378
6379
6380
6381
6382
6383
6384
6385
6386
6387
6388
6389
6390
6391
6392
6393
6394
6395
6396
6397
6398
6399
6400
6401
6402
6403
6404
6405
6406
6407
6408
6409
6410
6411
6412
6413
6414
6415
6416
6417
6418
6419
6420
6421
6422
6423
6424
6425
6426
6427
6428
6429
6430
6431
6432
6433
6434
6435
6436
6437
6438
6439
6440
6441
6442
6443
6444
6445
6446
6447
6448
6449
6450
6451
6452
6453
6454
6455
6456
6457
6458
6459
6460
6461
6462
6463
6464
6465
6466
6467
6468
6469
6470
6471
6472
6473
6474
6475
6476
6477
6478
6479
6480
6481
6482
6483
6484
6485
6486
6487
6488
6489
6490
6491
6492
6493
6494
6495
6496
6497
6498
6499
6500
6501
6502
6503
6504
6505
6506
6507
6508
6509
6510
6511
6512
6513
6514
6515
6516
6517
6518
6519
6520
6521
6522
6523
6524
6525
6526
6527
6528
6529
6530
6531
6532
6533
6534
6535
6536
6537
6538
6539
6540
6541
6542
6543
6544
6545
6546
6547
6548
6549
6550
6551
6552
6553
6554
6555
6556
6557
6558
6559
6560
6561
6562
6563
6564
6565
6566
6567
6568
6569
6570
6571
6572
6573
6574
6575
6576
6577
6578
6579
6580
6581
658
```

Review of "Apple Graphics & Arcade Game Design"

If you are at all interested in Apple graphics, or writing animated hi-res games, this book is for you. Jeffrey Stanton, the author, may already be known to you. He is the editor of "The Book of Apple Software, and also has several Apple arcade games on the market. "Apple Graphics & Arcade Game Design" (AGAG) is 288 pages long, and retails for \$19.95. (I am selling it for \$18 plus shipping.) A coupon in the back enables you to purchase all of the source code shown in the book on diskette for only \$15.

There are two parts to the book: first, a thorough explanation of Apple graphics, with numerous examples in both Applesoft and assembly language; second, design and programming of all the parts of a working arcade game.

AGAG is written for the advanced Applesoft or beginning assembly language programmer. You learn about both lo-res and hi-res graphics at the assembly language level. You learn the fundamentals, and then proceed to program scene scrolling, page flipping, laser fire, bomb drops, explosions, scoring, and paddle control routines. Sorry, nothing much about sound generation.

AGAG's pages are divided into 8 chapters as follows:

1. (25 pages) Applesoft Hi-Res
2. (34 pages) Lo-Res Graphics
3. (17 pages) Machine Language Access to Applesoft
Hi-Res Routines
4. (23 pages) Hi-Res Screen Architecture
5. (36 pages) Bit-Mapped Graphics
6. (90 pages) Arcade Graphics
7. (44 pages) Games that Scroll
8. (5 pages) What Makes a Good Game

I noticed a few errors in the book: on page 149, flow chart lines are incorrectly drawn; on page 284, there is a large block of repeated text, and therefore possibly a missing block which should have been in the space. The word "initialize" is always incorrectly spelled "initilize". The index is very brief, only about 70 lines long; I believe it should be about 3 or 4 times longer to really help in locating items of interest.

Jeff does not seem to know about the existence of the S-C Macro Assembler. He repeatedly mentions the TED, Big-Mac, Merlin Assemblers, and occasionally refers to Lisa and DOS ToolKit. All the listings are in the Big-Mac format. You should have no trouble adapting them to the S-C format.

AGAG is an excellent tutorial. and includes many useful programs and ideas for anyone interested in Apple graphics. I heartily recommend the book, ranking it just under "Beneath Apple DOS" in importance and utility.

Quick Way to Write DOS on a Disk.....Bob Perkins
Tussy, OK 73088

I just received the July AAL and liked the little article on the "FILEDUMP" command. I had already done just about the same thing.

In fact, I make a lot of changes to DOS. Too many to POKE in every time I boot up. So I started looking around for a simple way to replace the DOS image on a disk without disturbing the programs already on it, and without using MASTER.CREATE. The July Call A.P.P.L.E. had a program to do it, only it seems much more complicated than my solution.

I used the S-C Macro Assembler to create a text file like this:

```
:1000  LOAD HELLO
:1010  POKE -21921,0:POKE -18448,0:POKE
-18447,157:POKE-18453,0:CALL-18614
:TEXT WRITE.DOS
```

Note the leading blank before the LOAD and the first POKE. It is there to leave room for Applesoft's "]" prompt.

Whenever I want to write the DOS image on a disk, I use the SHOW command to list out WRITE.DOS, and then trace over the two command lines from Applesoft. Presto-Changeo, a new copy of DOS goes out to the disk. I suppose you could even EXEC it, though I prefer to trace over it and haven't tried EXECing.

The LOAD HELLO is there to get the boot file name into DOS's filename buffer. You can use whatever filename you want, of course. POKE-21921 tell DOS that the last command was an INIT for its startup procedure (i.e. AA5F:00). POKE-18448 and -18447 start the write at 9D00 (B7F0:00 9D). POKR -18453 sets the expected volume number to zero, so a match to any volume will occur (B7EB:00). The CALL is to the "write DOS image" code inside DOS.

Correction

Last month I described the BIT instruction incorrectly. The next to the last paragraph on page 2 (in "Run-Anywhere Subroutine Calls") should read:

The BIT instruction copies bit 7 of \$FF58 into the N-status bit, and bit 6 into the Overflow status bit. This, in other words (since \$FF58 has \$60 in it) clears N and sets Overflow.

BIT does not affect Carry Status in any way. BIT also sets or clears the Z-status bit, according to the value of the logical product of the A-register and the addressed byte. If you want Z and/or N to be flags to the calling program, you will have to modify them after the BIT instruction.

One of the more common questions we get is: "How do I best use the .IN and .TF directives to handle very large programs?"

The main technique we use is the Assembly Control File (ACF), a short source file which is mostly made up of .IN statements to call the other modules. Here is an example, called SAMPLE.ACF:

```
1000      .IN SAMPLE.EQUATES
1010      .PG
1020      .IN SAMPLE.CODE.1
1030      .PG
1040      .IN SAMPLE.CODE.2
1050      .PG
1060      .IN SAMPLE.DATA
1070      .PG
```

SAMPLE.EQUATES is all the definitions for the program, SAMPLE.CODE.1 and SAMPLE.CODE.2 are the main body of the program, and SAMPLE.DATA contains all the variables and ASCII text. When you want to assemble the program, just LOAD SAMPLE.ACF and type MON C then ASM. The Macro Assembler will load each file and assemble it, in the order they are listed in the ACF. The "MON C" shows you the "LOAD file name" for each file, helping you to tell what's where.

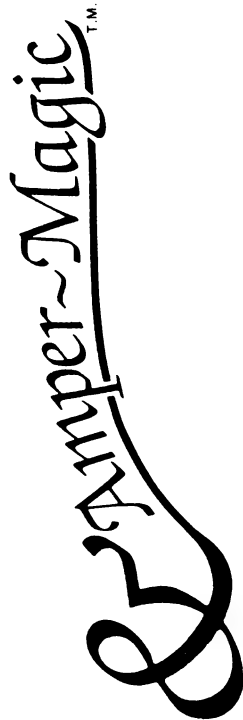
Using this technique, a program can conveniently be broken into as many modules as you want, and can be as large as you want. The Macro Assembler itself is 26 source files on two disks! To spread the files across more than one disk, just add drive (and/or slot) specifiers to all the file names.

You can also use the ACF to do global search-and-replace operations on the entire program. Here are the commands to search SAMPLE for all occurrences of the label MON.COUT:

```
:LOAD SAMPLE.ACF
:REP /      .IN/LOAD/A
:REP /      .PG/FIND "MON.COUT"/A
:TEXT COUT.SEARCH
:MON I
:EXEC COUT.SEARCH
```

This converts SAMPLE.ACF into an EXEC file that will list each occurrence of "MON.COUT" in every module of the program. Here's what the file looks like now:

```
1000 LOAD SAMPLE.EQUATES
1010 FIND "MON.COUT"
1020 LOAD SAMPLE.CODE.1
1030 FIND "MON.COUT"
1040 LOAD SAMPLE.CODE.2
1050 FIND "MON.COUT"
1060 LOAD SAMPLE.DATA
1070 FIND "MON.COUT"
```



MACHINE LANGUAGE SPEED WHERE IT COUNTS... IN YOUR PROGRAM!

Some routines on this disk are:

- Binary file info
- Delete array
- Disassemble memory
- Dump variables
- Find substring
- Get 2-byte values
- Gosub to variable
- Goto to variable
- Hex memory dump
- Input anything
- Move memory
- Multiple poke decimal
- Multiple poke hex
- Print w/o word break
- Restore special data
- Speed up Applesoft
- Speed restore
- Store 2-byte values
- Swap variables

For the first time, Amper-Magic makes it easy for people who don't know machine language to use its power! Now you can attach slick, finished machine language routines to your Applesoft programs in seconds! And interface them by name, not by address!

You simply give each routine a name of your choice, perform the append procedure once at about 15 seconds per routine, and the machine language becomes a permanent part of your BASIC program. (Of course, you can remove it if you want to.)

Up to 255 relocatable machine language routines can be attached to a BASIC program and then called by name. We supply some 20 routines on this disk. More can be entered from magazines. And more library disks are in the works.

These routines and more can be attached and accessed easily. For example, to allow the typing of commas and colons in a response (not normally allowed in Applesoft), you just attach the Input Anything routine and put this line in your program:

```
xxx PRINT "PLEASE ENTER THE DATE."; : & INPUT,DATES
```

&-MAGIC makes it Easy to be Fast & Flexible!

PRICE: \$75

Anthro - Digital Software
P.O. Box 1385
Pittsfield, MA 01202

&-Magic and Amper-Magic are trademarks of Anthro-Digital, Inc.
Applesoft is a trademark of Apple Computer, Inc.

The People - Computers Connection

The ACF is also a good place for the .OR and .TF statements, comments about the assembly process, and any condition flags. Here is a more complicated version of SAMPLE.ACF:

```

1000 *-----
1010 *          SAMPLE FILE TO DEMONSTRATE ACF
1020 *-----
1030 LC.FLAG .EQ 0  =0 IF UPPER CASE ONLY
1040 *          =1 IF LOWER CASE VERSION
1050 *-----
1060          .OR $803
1070          .DO LC.FLAG
1080          .TF B.SAMPLE.LC
1090          .ELSE
1100          .TF B.SAMPLE.UC
1110          .FIN
1120 *-----
1130          .IN SAMPLE.EQUATES
1140          .PG
1150          .IN SAMPLE.CODE
1160          .PG
1170          .DO LC.FLAG
1180          .IN SAMPLE.LOWER.CASE.ROUTINES
1190          .PG
1200          .ELSE
1210          .IN SAMPLE.NORMAL.ROUTINES
1220          .PG
1230          .FIN
1240          .IN SAMPLE.DATA
1250          .PG

```

To use this ACF, just LOAD it, EDIT line 1030 to set LC.FLAG to 0 or 1, set MON C, and ASM. The Macro Assembler will load the appropriate source files for the version you want and direct the object code to the correct target file. To turn this ACF into an EXEC file for searching, you must delete lines 1000-1120, 1170, 1200, and 1230 before doing the REP commands.

For more information on the .IN and .TF directives, see pages 4-6 and 5-3/4 in the Macro Assembler manual. Conditional assembly is discussed on pages 5-9/10 and in chapter 7.

Another Customizing Patch for the S-C Macro Assembler

Version 4.0 of the S-C Assembler stopped after any assembly error. Many users requested that I modify it to continue to the end of assembly, and display the error count at the end. So I did.

Now some users are requesting that I change it back. They walk away during assembly, and the error messages scroll off the screen. (But you can put .LIST OFF at the beginning, and then only the error lines will list.)

There is a very simple patch for this. The byte at \$1D6F (\$DD6F in the language card version) is now \$18. Change it \$38 and assembly will stop after the first error message.

Patch for S-C Macro Assembler.....Bob Sander-Cederlof

When I added the lower-case options to the S-C Macro Assembler, I overlooked the fact that within .AS and .AT strings, and in ASCII literal constants, you would want lower case codes to be assembled. The assembler as it now is converts all lower case codes to upper case during assembly. For example, ".AS /Example/" would assemble all upper case ASCII, just as though you had written ".AS /EXAMPLE/"

The following patches will correct this problem, allowing you to specify lower case strings and constants when you wish.

\$2961:EA EA EA EA EA EA

\$31B8<1235.124BM

\$1074:B8 31

\$118C:B8 31

\$11B2:B8 31

\$187F:B8 31

\$23FA:B8 31

\$31CF:C8 84 7B C9 60 90 04 29 5F 85 61 60

\$1240:20 CF 31

BSAVE ASM.WITH.LC.IN.AS,A\$1000,L\$21DB
(or whatever file name you wish)

The patches above are for the version which runs in mother-board RAM. The Language card version has different addresses, and you must first write-enable the language card. Assuming you are currently running the language card version, perform the patch as follows:

\$C083 C083

\$EAAD:EA EA EA EA EA EA

\$F304<D235.D24BM

\$D074:04 F3

\$D18C:04 F3

\$D1B2:04 F3

\$D87F:04 F3

\$E546:04 F3

\$F31B:C8 84 7B C9 60 90 04 29 5F 85 61 60

\$D240:20 1B F3

BSAVE LC.ASM.WITH.LC.IN.AS,A\$D000,L\$2327
(or whatever file name you wish)

Be aware that the above patches may conflict with other patches you may already have applied to your copy of the assembler. If you have already used the area from \$31B8 through \$31DB, or \$F304 through \$F326, you will need to use a different area and change the references accordingly.

Blinking Underline Cursor Routine.....Bill Linn

Early users of the ES-CAPE Applesoft Editing system (formerly known as AED II) have really come to appreciate the blinking underline cursor -- it simply doesn't tire the eyes as much as the standard flashing blank does. With the following subroutine, you can add this special touch to your own assembly language or BASIC programs!

The subroutine hooks into the monitor keyboard input vector at \$38 and \$39. Each time the monitor RDKEY subroutine is called, my KEYIN subroutine gets control. If the character on the screen at the cursor position is not an underline, I alternate the display of an underline and the original character every 1/4 second. If the original character was an underline, I alternate it with a blank. (If I alternate an underline with an underline, it is difficult to see anything happen!)

Lines 1210-1250 store the KEYIN subroutine's address in the keyboard input vector. When a request for a key press is made by an Applesoft INPUT command, for example, we get control at line 1270. The A-register has the current screen character. I save the A- and X-registers, because KEYIN must exit with the original values unchanged.

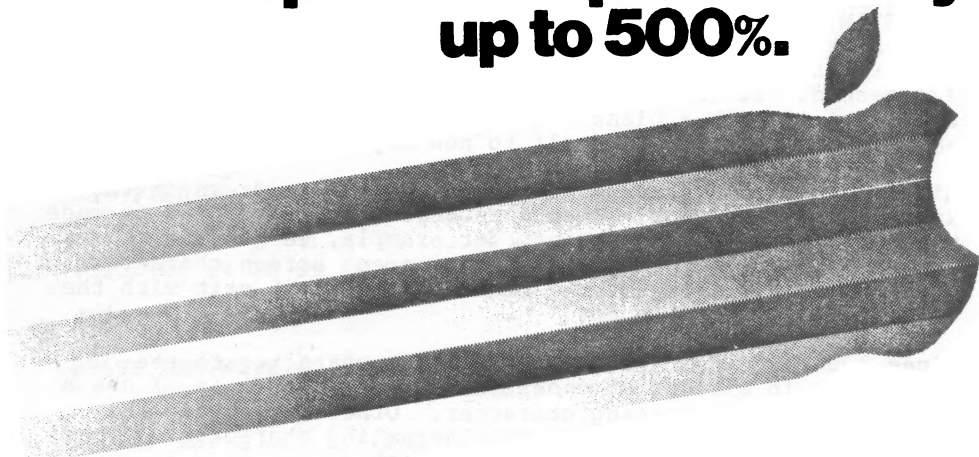
Lines 1290-1320 test the current screen character to see whether it is already an underline or not. If it is, I use a blank for the alternating character. Otherwise, I use the original screen contents for an alternating character. I push the alternating character onto the stack.

Lines 1330-1500 do the alternating. I look at the character on the screen: if it is an underline, I substitute the alternating character; if not, I store an underline. The lines 1430-1500 delay for about 1/4 second before the next alternation. If a keypress occurs, the loop ends by branching to ".5" at line 1540. You may wish to vary the blink rate by changing the value loaded into the Y-register at line 1430.

When a key is pressed we end up at line 1540, where I pop the alternating character off the stack. The I call the monitor bell subroutine for a short (10 half-cycles) bell. This makes an audible "click" for user feedback. (If you don't appreciate clicking keyboards, just delete lines 1550 and 1560.) Then I restore the Y-, X-, and A-registers to their original values, and jump into the monitor's KEYIN subroutine at \$FD26. The monitor restores the original character to the screen, and returns with the keypress value in the accumulator.

I have set the subroutine origin to \$300, but you can assemble it anywhere you like. In fact, it will run anywhere you put without reassembly, just so you load the correct address into \$38 and \$39 in the HOOK routine.

S&H Software presents Apple II™ users with two chances to increase speed and productivity up to 500%.



Universal Boot Initializer 4.0 will work up to 500% faster than standard Apple DOS 3.3...\$69.95

The new UBI 4.0 now includes The DOS Enhancer, a DOS-transparent routine that allows execution of Apple DOS 3.3 files (Integer, Applesoft and Binary) up to 500% faster than standard Apple DOS—depending on file length. In addition, a new "FREE" command in DOS now allows determination of free space on a disk in "any slot, any drive"—from the command mode or the program mode.

UBI 4.0-created disks increase efficiency by breaking the "language barrier" between Apple II hardware and software and breaking the "time barrier" by loading the RAM card with FPBASIC/INTBASIC (or leading assemblers) in 1.7 seconds. This unique combination of features greatly increases productivity: copyable disks, one-stage booting with DOS 3.3 or DOS 3.2.1 PROMS, fast-loading the RAM card with the "missing BASIC" (or your favorite utility), fast-(B)RUNning or fast-(B)LOADing your programs and complete compatibility with all DOS 3.3 programs.

The new UBI 4.0 package includes the utility disk, training disk, support disk, demo disk and complete documentation. System requirements: Apple II or Apple II Plus, ROM or RAM card, DOS 3.3 or 3.2.1 and one or more disk drives.

Amper-Sort/Merge (A-S/M) works up to 500% faster than even VisiCorp's VisiFile™ program...\$52.95

The fastest "file clerk" you've ever met. Of all the sort utilities developed to manage Apple II data files, none does the job nearly so fast as Amper-Sort/Merge.

Here's a quick profile of "A-S/M": With 25K of working memory, one of five unsorted files can be sort/merged into a single file of up to 125K per disk. If a file to be sorted is more than 25K in length, the utility temporarily lays it aside to be sorted and merged when more memory space is available.

Because sorting routines take up to 50% of the computer running time in many business applications, you'll reap continuing benefits having this "invisible speed demon" on your Apple II team. We estimate that it will save twenty to thirty minutes a day of your "human" clerk's time—time that would otherwise be spent waiting for "sort/merge" operations.

The A-S/M "speed demon" package includes the utility disk, the training disk and 24 page instruction manual. System requirements: 48K Apple II, ROM or RAM card, DOS 3.3 and one or more disk drives or 48K Apple II Plus, DOS 3.3 and one or more disk drives.

To Order: Send Check To S&H Software, Box 5, Manvel, ND 58256 Credit Cards: Phone Cybertronics International Clearinghouse at 212 532-3088.

S&H Software



Box 5 Manvel ND 58256
(701) 694-2574

After assembly, assuming it is originated at \$300, you can BSAVE it with "BSAVE B.UNDERLINE,A\$300,L\$3C. Then to activate this routine from Applesoft, just BRUN the file B.UNDERLINE. All keyboard input through the standard monitor RDKEY subroutine (\$FDOC) or Applesoft GET and INPUT statements will be prompted by the underline cursor. An "IN#0" will restore the familiar flashing blank. Have fun!

```

1000 *SAVE S.UNDERLINE CURSOR
1010 *-----
1020 *      BLINKING UNDERLINE CURSOR
1030 *      WRITTEN BY BILL LINN
1040 *-----
1050 *      .OR $300
1060 *-----
0024- 1070 MON.CH      .EQ $24
0028- 1080 MON.BASL   .EQ $28
0038- 1090 MON.KSWL   .EQ $38
004E- 1100 MON.RNDL   .EQ $4E
1110 *-----
FBE4- 1120 MON.BELL2   .EQ $FBE4
FCA8- 1130 MON.WAIT    .EQ $FCA8
FD26- 1140 MON.KEYIN3  .EQ $FD26
1150 *-----
00A0- 1160 BLANK      .EQ $A0
00DF- 1170 UNDERLINE .EQ $DF
1180 *-----
C000- 1190 KEYBOARD   .EQ $C000
1200 *-----
0300- A9 09 1210 HOOK    LDA #KEYIN   SET INPUT HOOK
0302- 85 38 1220        STA MON.KSWL
0304- A9 03 1230        LDA /KEYIN
0306- 85 39 1240        STA MON.KSWL+1
0308- 60    1250        RTS
1260 *-----
0309- 48    1270 KEYIN   PHA          SAVE SCREEN CHAR
030A- 86 4E 1280        STX MON.RNDL   SAVE X-REG
030C- C9 DF 1290        CMP #UNDERLINE IF CHAR ON SCREEN IS
030E- D0 02 1300        BNE .1        AN UNDERLINE
0310- A9 A0 1310        LDA #BLANK    THEN ALTERNATE WITH BLANK
0312- 48    1320        PHA          SAVE CHAR TO ALTERNATE
1330 *-----
1340 *      ALTERNATE UNTIL KEY IS PRESSED
1350 *-----
0313- A9 DF 1360 .2     LDA #UNDERLINE
0315- A4 24 1370        LDY MON.CH
0317- D1 28 1380        CMP (MON.BASL),Y
0319- D0 02 1390        BNE .3
031B- 68    1400        PLA          GET ALTERNATE CHAR
031C- 48    1410        PHA          MAINTAIN ON STACK ALSO
031D- 91 28 1420 .3     STA (MON.BASL),Y
031F- A0 50 1430        LDY #80     80*256 BETWEEN BLINKS
0321- AD 00 C0 1440 .4     LDA KEYBOARD KEY PRESSED?
0324- 30 08 1450        BMI .5        YES, CLICK AND RETURN
0326- CA    1460        DEX
0327- D0 F8 1470        BNE .4
0329- 88    1480        DEY
032A- D0 F5 1490        BNE .4
032C- F0 E5 1500        BEQ .2        ...ALWAYS
1510 *-----
1520 *      A KEY HAS BEEN PRESSED
1530 *-----
032E- 68    1540 .5     PLA          POP STACK ONCE
032F- A0 0A 1550        LDY #10     MAKE A "CLICK"
0331- 20 E4 FB 1560        JSR MON.BELL2
0334- A4 24 1570        LDY MON.CH
0336- A6 4E 1580        LDX MON.RNDL   RESTORE X-REG
0338- 68    1590        PLA          RESTORE ORIGINAL SCREEN CHAR
0339- 4C 26 FD 1600        JMP MON.KEYIN3

```

I had already started writing my own debugger when I discovered QUICKTRACE; it was just what I needed and saved me all that work.

It has a good display that does not interfere with the normal Apple text screen. You can even trace code that sets the KSWL and CSWL switches and outputs to the screen. The tracing display takes the bottom four lines, but pressing the "P" key causes the normal bottom four lines to be displayed.

Tracing can be in one of three modes: single-step, trace, and background. Single-step and trace are what you would expect, analogous to the commands in the old Apple monitor ROM. Background turns off the display of executed instructions until a breakpoint occurs or the "ESC" key is pressed. This makes background the fastest mode.

Breakpoints can be set to stop when:

1. Any register or a memory location takes on a specified value.
2. An address or a range of addresses is referenced.
3. A specified opcode occurs.

QUICKTRACE can be BRUN at any point in memory and then called from your code by a JSR, or you can preset the QUICKTRACE program counter and start tracing at any location.

Subroutines can be executed at full 6502 speed (not traced). If you already know what the subroutine does there is no need to trace through it. Normally DOS calls are automatically done this way to prevent timing problems.

Overall I feel that QUICKTRACE is one of the five or so best programs I have ever purchased and no machine code programmer should be without it.

One feature not to be overlooked: QUICKTRACE is not copy protected.

QUICKTRACE was programmed by John Rogers and it is distributed by Anthro-Digital Software (formerly called Aurora Systems). It only costs \$50.

Current Advertising Rates

For the September 1982 issue the price will be \$60 for a full page, \$35 for a half page. To be included, I must receive your camera-ready copy by August 20th.

Apple Assembly Line is published monthly by S-C SOFTWARE CORPORATION, P. O. Box 280300, Dallas, Texas 75228. Phone (214) 324-2050. Subscription rate is \$15 per year in the USA, sent Bulk Mail; \$18 per year sent First Class Mail in USA, Canada, and Mexico; \$28 per year sent Air Mail to other countries. Back issues are available for \$1.50 each (other countries add \$1 per back issue for postage). All material herein is copyrighted by S-C SOFTWARE, all rights reserved. (Apple is a registered trademark of Apple Computer, Inc.)